Downloaded from orbit.dtu.dk on: Dec 18, 2017

Technical University of Denmark

Defining Interactions and Interfaces in Engineering Design

Parslov, Jakob Filippson; Mortensen, Niels Henrik; Hvam, Lars; Bysted, Tommy

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

Link back to DTU Orbit

Citation (APA):
Parslov, J. F., Mortensen, N. H., Hvam, L., & Bysted, T. (2016). Defining Interactions and Interfaces in Engineering Design. Technical University of Denmark (DTU). (DCAMM Special Report; No. S200).

DTU Library
Technical Information Center of Denmark

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

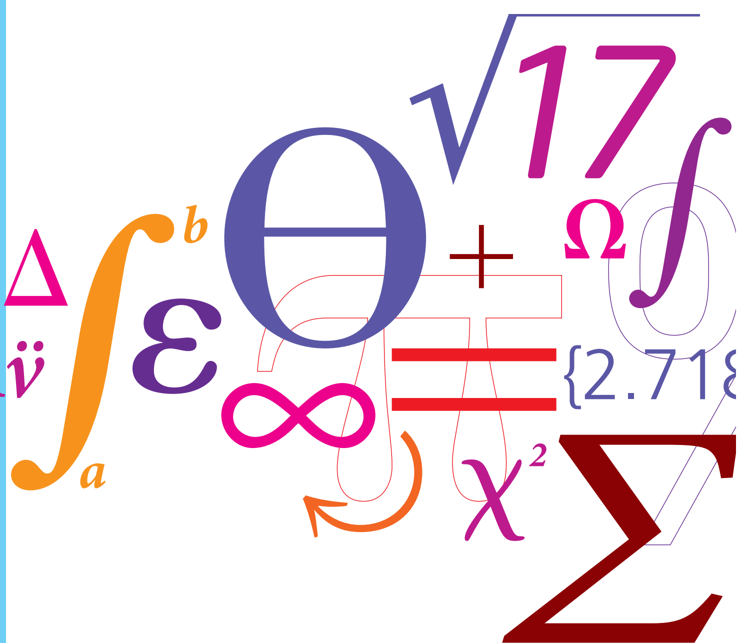
- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Defining Interactions and Interfaces in Engineering Design

PhD Thesis

$$(EIv'')'' = q - \rho A\ddot{v}$$



Jakob Filippson Parslov

DCAMM Special Report No. S200

March 2016

DTU Mechanical Engineering

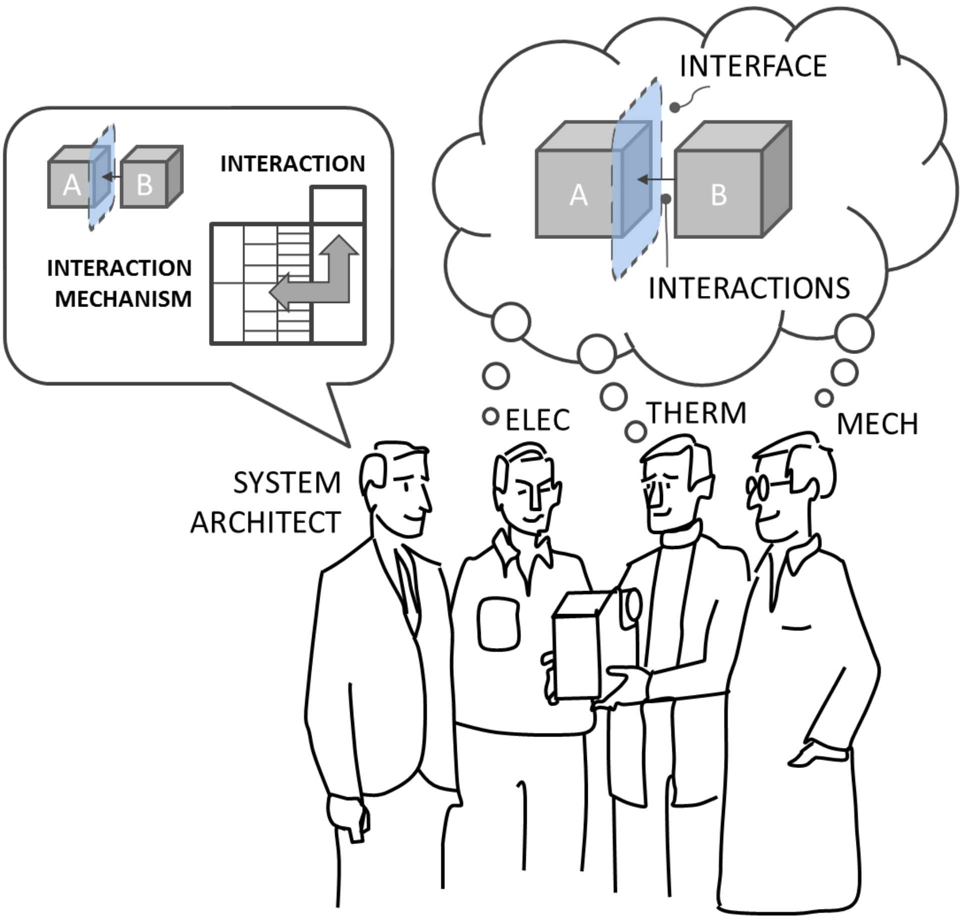
Department of Mechanical Engineering

Thesis for the degree of Doctor of Philosophy

Defining Interactions and Interfaces in Engineering Design

by

Jakob Filippson Parslov



Department of Mechanical Engineering
Technical University of Denmark
March, 2016

Technical University of Denmark

Defining Interactions and Interfaces in Engineering Design

Jakob Filippson Parslov

Section of Engineering Design and Product Development
Department of Mechanical Engineering
Technical University of Denmark
Produktionstorvet, Bld. 426 B
DK-2800 Kgs. Lyngby
Denmark

PhD Thesis

ISBN: 978-87-7475-447-3
DCAMM Special Report No.: S200

© Copyright by Jakob F. Parslov
All rights reserved
2016

Supervisor
Niels Henrik Mortensen, PhD
Head of Section, Professor
Section of Engineering Design and Product Development
Department of Mechanical Engineering
Technical University of Denmark

Co-supervisor
Lars Hvam, PhD
Professor
Operations Management
Department of Management Engineering
Technical University of Denmark

Company supervisor
Tommy Bysted, PhD
R&D Director
New Product Development
Radiometer Medical ApS

Opponents
Johan Malmqvist, PhD
Dean of education MATS, Chair Professor
Product and Production Development
Chalmers University of Technology

Christian Weber, PhD
Dean of Department of Mechanical Engineering
Technical University of Ilmenau

Michael Wörösch, PhD
Director of Systems Engineering
GN Resound

This PhD thesis was publicly defended on June 3rd, 2016

Abstract

This PhD thesis focuses on the understanding and definition of interactions and interfaces during the architectural decomposition of complex, multi-technological products. The *Interaction and Interface Framework* developed in this PhD project contribute to the field of engineering design research.

Developing complex, multi-technological products involves the joint effort of multiple engineering disciplines in order to arrive at an end product, which satisfies its requirements. A major challenge is however the fact that bringing together engineers from different technical backgrounds means that they have different conceptual viewpoints on the product and use different 'technical languages' to communicate. Some terms like an *interface*, is used frequently in engineering however with no commonly declared meaning and is thus subject to much interpretation across engineering disciplines. It is well-known that most problems arise at the interfaces during product development, which is why there is a need for a rigorous and multi-disciplinary treatment of the concept of *interfaces* as well as *interactions*.

On the basis of a two-year case study at a medical device manufacturer, the role of interactions and interfaces in product family development has been investigated. The case study showed that for this particular case, interaction and interface descriptions represents the rationales needed to reuse documentation across multiple product variants. The interaction and interface descriptions thus become documents of legal matter and must therefore be unambiguously and completely described.

Following this observation, a comprehensive and systematic literature review has been performed in order to investigate the definition and perception of an interface. The review resulted in a classification revealing 13 dominant perceptions of what an interface is from an academic perspective including the observation of an apparent confusion between the terms *interaction* and *interface*. In addition, a case example of a solenoid valve was examined in order to reason out the likely causes of problems occurring at interfaces. The case example showed that interfaces that reside at the boundary between engineering disciplines are vulnerable to misinterpretation and rework.

Based on this understanding, this thesis presents a first principles, physics-based *Interaction and Interface Framework*, which provides a 'common language' across any engineering discipline for describing and communicating about interactions and interfaces in engineering design. The framework contains classifications of three key terms; interaction, interaction mechanism, and interface. Due to the first principles, physics-based approach to deriving the framework, it has been possible to arrive at a classification of interaction mechanism, which is *mutually exclusive* (no overlap) and *collectively exhaustive* (no gaps). This contribution changes the existing paradigm of reasoning about interactions and allows for an unambiguous architectural decomposition of a product.

The framework further proposes an 8-step architecting approach explicitly articulating how to systematically apply the framework top-down thus enabling complete and unambiguous descriptions of interactions and interfaces throughout the system. A tool called an Interaction Specification Wheel (ISW) is introduced to support consistency in writing requirements and specifications. All of the contributions have been evaluated in an initial test, which indicated a positive effect on their ability to capture interactions and unambiguously specify them. Further research is needed to obtain statistical significance.

Future research may investigate how to incorporate the framework into practice and further evaluate the high level effects. This will most likely require two or more case studies in real-life projects.

Resume

Denne PhD afhandling omhandler forståelsen og definitionen af interaktioner og interfaces ved arkitekturnedbrydning af komplekse og multi-teknologiske produkter. *Interaktions- og Interface Frameworket,* som er fremkommet af dette forskningsprojekt, bidrager således til *Engineering Design* forskningsfeltet

For at kunne udvikle komplekse og multi-teknologiske produkter som opfylder dets krav om ønsket funktionalitet og performance, er det nødvendigt med et produktivt samarbejde mellem forskellige ingeniørdiscipliner. En hovedudfordring er dog, at der ikke findes et fælles sprog for visse begreber såsom et *interface* (dvs. en grænseflade), hvilket resulterer i at forskellige ingeniørdiscipliner anvender deres egen fortolkning af begrebet. Alvoren af dette forstærkes yderligere af, at interfaces er det sted i produktet, hvor flest problemer opstår under udviklingen, hvorfor der er et udtalt behov for en stringent og multidisciplinær behandling af begrebet *interface* såvel som det nært beslægtede begreb *interaction*.

Den overordnede rolle af interaktions- og interfacebeskrivelser er blevet undersøgt i et toårigt casestudie hos en medikoudstyrsproducent. Casestudiet viste, at i dette specifikke tilfælde, bliver interaktions- og interfacebeskrivelserne brugt som rationaler for genbrug af dokumentation på tværs af produktvarianter. Interaktions- og interfacebeskrivelserne bliver således ophøjet til dokumenter af juridisk karakter, hvorfor de nødvendigvis bør fremstå helt entydige og komplette.

Som følge af denne observation blev der foretaget et omfattende og systematisk litteraturstudie for at undersøge definitionen og opfattelsen af et *interface*. Reviewet resulterede i en klassifikation med 13 forskellige opfattelser af hvad et *interface* er fra et akademisk perspektiv, herunder en erkendelse af en sammenblanding mellem begreberne *interaktion* og *interface*. Derudover blev et case eksempel af en solenoid ventil brugt til at udlede de sandsynlige årsager til at problemer opstår ved interfaces. Case eksemplet viste, at interfaces, som er placeret i grænsefeltet mellem tekniske discipliner, er følsomme overfor fejlfortolkning og bør håndteres intensivt.

For at adressere de identificerede fænomener introducerer denne PhD afhandling et *first principles, fysikbaseret Interaktions- og Interface Framework* som giver et fælles sprog på tværs af alle ingeniørdiscipliner til at beskrive og kommunikere omkring interaktioner og interfaces indenfor *Engineering Design*. Frameworket indeholder klassifikationer af tre nøglebegreber; *interaktion*, *interaktionsmekanisme* og *interface*. Som følge af en first principles, fysik-baseret tilgang til at udlede frameworket har det været muligt at komme frem til en klassifikation af interaktionsmekanismer som er *gensidig uafhængig* (ingen overlap) og *helt komplet* (ingen mangler). Dette framework ændrer det eksisterende paradigme omkring at ræsonnere om interaktioner og interfaces og tillader en entydig nedbrydning af arkitekturen af et produkt.

Frameworket bidrager derudover med en 8-trins arkitekturtilgang, som eksplicit adresserer, hvordan frameworket kan blive taget systematisk i brug top-down, og derigennem tillade komplette og entydige beskrivelser af interaktioner og interfaces i systemet. Desuden introduceres der et værktøj kaldet *Interaction Specifications Wheel (ISW)* til at understøtte konsistens i beskrivelsen af interaktions- og interface-krav og specifikationer. Alle bidrag i afhandlingen er blevet evalueret med indledende tests som indikerer en positiv effekt på testdeltagernes evne til at identificere og beskrive interaktioner. Det vil kræve yderligere undersøgelser at opnå statistisk signifikans.

Fremtidig forskning ved brug af casestudier bør undersøge, hvordan *Interaktions- og Interface Frameworket* kan adopteres i praksis og yderligere evaluere de overordnede effekter af anvendelsen.

Preface

This PhD thesis presents the results from an Industrial PhD project carried out in the period from January 2013 to March 2016. The Industrial PhD project was co-financed by *The Agency for Science, Technology, and Innovation* under *Ministry for Higher Education and Science* as well as by *Radiometer Medical ApS.* Sincere thanks to both the agency and Radiometer Medical ApS for granting me this opportunity. The project was executed in collaboration with *Section of Engineering Design and Product Development, Department of Mechanical Engineering, Technical University of Denmark.*

The thesis is paper-based meaning that the primary purpose of this thesis is to tie together the research results into a coherent story and argue why we should believe in these research results.

Over the course of ~38 months I have interacted with many highly intellectual and nice people, from which I have been *motivated*, *inspired*, *challenged*, *pushed*, *and mentally supported*.

Many thanks to Professor Niels Henrik Mortensen for continuous support and for believing in my abilities from day one and throughout the project. Also thanks to my co-supervisor Professor Lars Hvam for stepping in with concrete and valuable second opinions when needed.

Many thanks to my company supervisor, R&D Director Tommy Bysted, for your valuable supervision on both the project management aspects as well as the content. I have grown both personally and professionally from your input. Sincere thanks to senior specialist at Radiometer Medical ApS, Barney Gerrard, for being an invaluable discussion partner. The rigor and quality of this project have improved significantly from your input.

A special thanks to Professor Emeritus Mogens Myrup Andreasen for countless inspiring discussions. At our first meeting you argued that the best way to learn is through *dialog*. Having been exposed to your heuristic, I completely agree.

Thank you to Prof. Olivier de Weck from Department of Astro- and Aeronautics Department, MIT, for giving me the opportunity to stay at your lab and discuss my research with you and your co-workers - truly a great experience.

Also thanks to all of my colleagues at both DTU and Radiometer for inspiration, motivation and laughs.

Finally, a special thanks to my family for your moral support throughout the project and to my wonderful girlfriend Laura, for her love, patience and understanding. You have supported me in the best way possible, thank you!

Jakob Filippson Parslov

Copenhagen, March 2016

List of appended papers

Paper A (Published, recommended for journal publication, honors)

Jensen, T. V., Parslov, J. F., & Mortensen, N. H. (2015, August). Enabling Reuse of Documentation in New Medical Device Development: A Systematic Architecting Approach. In *ASME 2015 IDETC/CIE DTM Conference* (pp. V007T06A024-V007T06A024). American Society of Mechanical Engineers

Paper B (Published)

Parslov, J. F., & Mortensen, N. H. (2015). Interface definitions in literature: A reality check. *Concurrent Engineering*, 1063293X15580136

Paper C (Submitted, currently under peer review)

Parslov, J. F., Gerrard, B., & Mortensen, N. H. (2016). Understanding Interactions in Complex Multi-Technological Products – A First Principle, Physics-based Theoretical Framework, *Research in Engineering Design*

Paper D (Submitted, currently under peer review)

Parslov, J. F., Gerrard, B., & Mortensen, N. H. (2016). Defining Interactions and Interfaces in Complex Multi-Technological Products – A Multi-disciplinary, Physics-based Approach. *Research in Engineering Design*

Table of contents

1 Introduction

The following PhD thesis marks the end of a three-year research project within the field of engineering design – more specifically design of complex multi-technological products or systems. This first part of the thesis will feature an introduction to the background, and problem area followed by research objectives, definitions of key terms, research questions, aim and scope of the thesis, and finally an outline of the thesis.

1.1 Background and problem area

1.1.1 Industry challenges

Two decades of globalization have meant that companies of today compete on a global scale with a very diverse set of customers. Customers expect more from their products in terms of quality and functionality causing the companies to race for superior functionality and performance of their offered products in order to improve competitiveness and increase market shares. To meet these demands from the market, companies increasingly integrate multiple technologies into their products causing the product complexity to rise. Also, with technologies often being multi-disciplinary (i.e. mechanics, electronics, software etc.) successful product development increasingly relies on an effective collaboration and communication between the various engineering disciplines involved. One of the challenges with a multi-disciplinary development environment is the fact that the mental models and conceptual viewpoint differ across disciplines, even within disciplines, which impedes communication and common understanding (Jarratt et al. 2004).

There is a common understanding in academia and industry that most problems occur at the *interfaces* during product development (Grady 1994; Kapurch 2007; Wheatcraft 2010; Buede 2012). There are likely many causes behind this ascertainment, but one reason may be the way *roles* and *responsibilities* are arranged around the product development activities coupled with a lack of common language (Parslov and Mortensen 2015).

In many companies development teams are organized in a matrix-formation around a structural decomposition of a product with module owners and technical leads governing each technical discipline across (Ulrich and Eppinger 2012). In complex integral products however, functions and properties may not be isolated to a single module but rather span the structural composition of the product and thus span areas of ownership. This means that a *change*, which is induced in one area of the product, in one discipline, may propagate to other modules in other disciplines because they are related by function. See Figure 1.

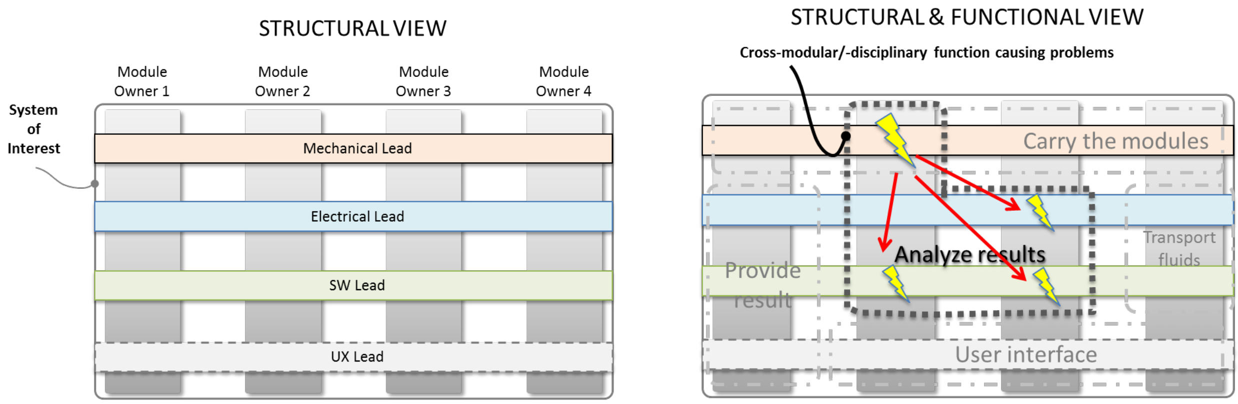


Figure 1 A structural view (left) showing the allocation of ownership with clearly separated areas of responsibility. There may however be a discrete coupling between the modules and disciplines from a functional perspective. A 'lightning' indicates how a change propagates through the system and leads to necessary changes in other modules/disciplines

As we describe in Parslov & Mortensen (2015), there may therefore be interfaces which become highly critical because they reside at the boundary between modules and disciplines at the same time. They may therefore have a big impact on the overall system functionality if not managed intensely while also being vulnerable to misinterpretation due to *lack of common language* between the technical disciplines.

The significance of this issue is further amplified by the fact that the later in the product development phase an interface problem is detected, the higher the cost of removing the defect. See Figure 2.

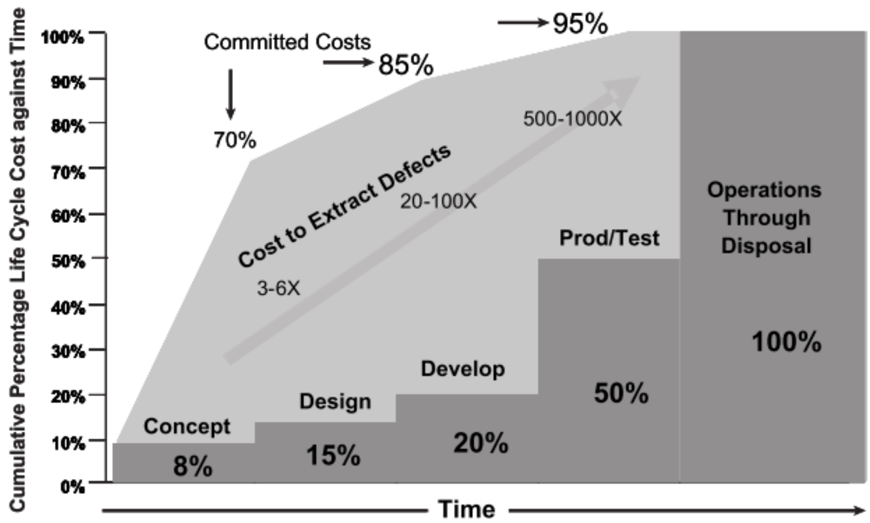


Figure 2 Committed life cycle cost against time. Cost associated with extracting defects increases exponentially with time. Adapted from (Haskins et al. 2006)

Some of the key issues related to interfaces in industry are:

- Lack of common understanding across various engineering disciplines about how to describe the multi-technological interfaces properly in the early architectural phases (Parslov and Mortensen 2015)
- Issues with interface compatibilities, which are discovered very late in a project is a major cause of cost overruns and product failure (Wheatcraft 2010)

- Detrimental emergent behavior arises due to complex interaction patterns which are difficult to predict and capture before the product is integrated, built and tested (Ulrich and Eppinger 2012)
- Only very few engineers have the capacity to engage in detailed technical discussions across various technical disciplines, while also abstracting from the details and reason on a systems level (Jarratt et al. 2004)
- It is difficult for companies to develop a system solution on a conceptual level, which needs to work well together with the surrounding systems and comply with all requirements – without being able to do any detailed analysis. Kihlander & Ritzén (2012) describes this phenomenon as achieving *compatibility before completeness*
- No clear functional ownership means that nobody takes responsibility for cross-modular or cross-disciplinary issues – e.g. tolerance chain issues with mechanical components and electrical sensors involved, EMC issues etc.

In some complex product domains, like astro- and aeronautics, automotive, there has been quite some focus on *system integration* and *systems engineering*, with *interface management* as a key product development activity (Lalli et al. 1997; Wheatcraft 2010; ECSS 2015; Yasseri 2015). Some of the methods, templates and standards that are available are *Interface Control Documents (ICD)*, *Interface Definition Documents (IDD)*, *Interface Requirements Documents (IRD)*, *ECSS Interface Management Standard* etc. The focus of these documents is however not on the early architectural phases and does not seek to create a common understanding and language of interfaces across engineering disciplines. They provide a standardized template for filling in requirements and specifications, but do not prescribe what and how to specify the content of it (Rahmani and Thomson 2012) and may be domain specific (i.e. ECSS Interface Management Standard (ECSS 2015) addresses space systems). A major reason for requirements modifications during the early stages of complex systems development has to do with *incomplete capture*, *traceability issues*, and *incorrect* or *ambiguous language* (Fernandes et al. 2014).

Another observed trend, which calls for rigor concerning interfaces is the transformation in industry from developing stand-alone products to developing families of products based on a platform (Harlou 2006; Simpson et al. 2006; Ulrich and Eppinger 2012; Simpson et al. 2014). The benefits of developing a family of products is the potential to gain synergies across the family; production volume of certain components may increase thus lowering the production cost (i.e. economies of scale), customers may familiarize better with the company brand values, quality can be matured over a longer time period through reuse of modules, thus lowering the number of customer complaints etc. However, this transformation is not without its challenges:

- Interfaces to the platform must be compatible with future modules which have not yet been conceived
- Failures at platform interfaces, are high-risk because they may propagate to all product variants which are based on the same platform
- A platform has a long life-span, meaning that, interfaces to the platform may need to be designed with excess capacity or flexibility in order to be able to cope with future high-performing modules (Suh et al. 2007). The cost of the first platform-variant is therefore proportionately higher than the future variants. Being loyal to the platform all the way to the last variant requires top management commitment

1.1.2 Academic challenges

The multi-technological nature of today's products challenge academia to come up with theories, methods, and tools that are not confined by traditional technical disciplines such as mechanical engineering, electrical engineering, software engineering etc. (Torry-Smith 2013). Newer areas of research such as *Systems Engineering* (Sage and Cuppan 2001; Haskins et al. 2006; Kapurch 2007; Dickerson and Mavris 2010; Standard and ISO/IEC/IEEE 2011; Bonnema et al. 2015) have emerged, which seek a multi-disciplinary approach to engineering design. *Systems Engineering* however focuses primarily on the processes involved in developing a product, and not on the nature of the product itself (the object). This aspect is important because a condition for an effective process is that there is an unambiguous understanding of the object. *Systems architecting* (Crawley et al. 2004; Hölttä-Otto et al. 2014; Crawley et al. 2015) represents an early-stage Systems Engineering activity, which deals with functional and physical decomposition of systems and thus tries to remain rather discipline independent. However, the theoretical contributions regarding functional interactions and interfaces are inconsistent from a physics perspective and seem to be developed on the basis of convenience to existing practices (Andreasen 1980; Hubka and Eder 1988; Wie et al. 2001; Dickerson and Mavris 2010; Zheng et al. 2016).

Some key issues related to interfaces in academia are:

- No clear understanding of what an interface is in multi-technological products
- The terms *interface* and *interaction* are used interchangeably
- Much research on interfaces within engineering design is mono-disciplinary
- Much research on interfaces within engineering design is empirically based and is therefore difficult to verify and validate
- Current classifications of interfaces as functional interactions are inconsistent from a physics perspective

Having reviewed a substantial number of papers on interfaces in engineering design, it is the author's opinion that there is a tendency for much of the contributions in this area, to be based on *convenience* or *practicalities* in order to fit the language and conceptual mindset in the intended context. The risk is a lack of internal consistency and generality, thus leaving room for interpretations and misuse.

There is hence a need from both industry and academia to further investigate the nature of interactions and interfaces in multi-technological products by means of a rigorous research method that enforces a consistent, generalizable result, applicable to the context in which it is used. One such approach could be deductive *first principles* reasoning, where the point of reasoning starts at the established fundamental 'truths' of physics. This approach has been adopted in this research and will be described in section 2.2 Research Methodology.

1.2 Aim and objectives of the research

The *overall aim* of this project is to support *early-stage architecture based product development* through an explicit focus on *how to conceptually understand and model interactions and interfaces across any engineering discipline*.

This industrial research project can be characterized as *applied research*, due to the clear connection between the real-world practical problems and the theoretical problems. The project also has a clear link to more basic research such as physics in order to arrive at prescriptive support, which is rigorous and scientifically sound from both physics as well as an engineering design point of view.

The project has several *objectives*, which may be categorized according to *theoretical* and *practical* applications.

Theoretical objectives:

- To expand the body of knowledge of *Engineering Design* research and *Systems Engineering* by providing knowledge about the existing paradigm concerning *interactions and interfaces*
- To expand the body of knowledge about reasoning in multi-disciplinary *Engineering Design* research

Practical objectives are to *increase product development efficiency and effectiveness* by:

- *Reducing ambiguity* during early-phase architectural synthesis of multi-technological products by enhancing the knowledge of *interaction and interface*
- *Supporting communication* and *reasoning* across multiple engineering disciplines during product development by enhancing the knowledge of *interaction and interface*
- *Supporting completeness* in *interaction* and *interface descriptions* by enhancing the knowledge of *interaction and interface* in multi-technological, complex products

Both the aim and the objectives (theoretical and practical) will be addressed in the conclusion.

1.3 Definition of key terms used

The following key terms are used frequently during the thesis. I recognize that there may not be a 'single truth' to the definition of these terms, however, the chosen definitions are considered useful and consistent with the argumentation in this thesis. Definitions that have zero reference are defined by the author for the sole purpose of this research project. See Table 1. In addition, any capitalized terms in this thesis have been defined as part of this research. If a term is used in a non-capitalized format the term is merely a generic proxy, e.g. "interaction" is a proxy for INTERACTION and INTERACTION MECHANISM.

Table 1 List of definitions of key terms used throughout the thesis

Term or concept	Definition / Description
Complex system	A system with components, interactions, and interfaces that is difficult to describe, understand, predict, manage, design, or change. Inspired from (Weck et al. 2011)
Framework	A conceptual scheme of mental constructs, models, and definitions that collectively frame or describe one or more phenomena. Inspired from (OED 2015a)
FUNCTION	The purposeful transformation from input to output realized by a physical manifestation of the product. Each input and output from a system is an INTERACTION. It describes what the system *does,* and not how *good* it does it

Module	"A module is a unit whose structural elements are powerfully connected among themselves and relatively weakly connected to elements in other units. Clearly there are degrees of connection, thus there are gradations of modularity" (Baldwin and Clark 2000)
Multi-disciplinary	'Multi-disciplinary' is a characteristic of a design activity and applies whenever more than one engineering discipline is involved in the development of a product. Inspired from Torry-Smith (2013)
Multi-technological	'Multi-technological' is a characteristic of a product and refers to the fact that the constituent elements of the product, e.g. technologies, modules or components, are developed by multiple engineering disciplines, e.g. mechanical, electrical, software engineering. Inspired from Torry-Smith (2013)
Phenomenon	A fact or observed situation, where the cause or explanation is in question. Inspired from (OED 2015b)
Product architecture	"The arrangement of functional elements, the mapping from functional elements to physical components, and the specification of the interfaces among interacting components" (Ulrich 1995)
Property / Functional property	Related to the goodness, with which a system executes its function, e.g. reliability, accuracy, predictability etc. (Andreasen et al. 2015)

1.4 Research questions

The following research questions have been instrumental to the structuring and execution of this PhD project. They collectively frame the area of research ranging from exploratory, descriptive questions, to more prescriptive questions.

The first research question (RQ1) is about understanding the phenomena concerning interfaces in product family design practice. There has been quite a lot of research published on product family design where interfaces are highlighted as an important aspect to manage and define. RQ1 attempts to shed light on the medical device industry and the specific role of interfaces in an industry of heavy regulation. RQ1 therefore asks:

> RQ1
> *What is the high-level role of interactions and interfaces in product family design in new medical device development?*

To answer this question an empirical case study is undertaken at the case company Radiometer Medical ApS that manufactures medical devices (Yin 2013). The case study involves interviews with various domain expert as well as review of codified information from a multitude of product data sources and documentation files.

Following this investigation of the role of interactions and interfaces in practice, the next couple of questions are of a more theoretical character. It is the author's experience, based on numerous informal domain expert interviews that engineers from different technical backgrounds tend to have very different perception of what an interface is. Since problems often occur at interfaces it is worth investigating whether there is a discrepancy of perceptions from a literature point of view. RQ2 therefore asks:

RQ2

How are interfaces defined and perceived in literature?

The question is answered based on a comprehensive systematic literature review on the definition of an interface. As part of this treatment of the literature, it is investigated *why* problems might occur at interfaces. This is captured in RQ3:

RQ3

What phenomena in multi-disciplinary product development are likely causes of problems occurring at interfaces?

The answers to RQ3 are hypothesized in a discussion of the findings from the literature review and a case example. RQ1, 2, and 3 thus clarifies the phenomena concerning interfaces in product development. On the basis of these findings, the following research questions outline the prescriptive phase of the project.

One of the key findings is that ambiguity in the definition of interfaces may lead to discrepancies in the perception of interfaces across multiple disciplines thus risking miscommunication and rework, e.g. the inconsistent distinction between *interactions* and *interfaces*. Therefore, the aim of the prescriptive phase is to develop a theory, method and tool that may *reduce ambiguity* during the architectural decomposition of complex systems. However, a fair assumption in this project is that *it is not possible to define the concept of an interface, without knowing what is transferred across that interface, the interaction*. Therefore, it follows that we must investigate the nature of interactions before defining an interface. RQ4 asks:

RQ4

How can interactions be classified using a physics-based first principles approach?

As RQ4 suggests, the applied research approach is a first principles approach to fundamental physics, meaning that the theoretical contribution is deduced without any empirical assumptions from the very fundamentals. This allows for the creation of a *mutually exclusive* (no overlap) and *collectively exhaustive* (no gaps) classification concerning interactions, which is unambiguous and multi-disciplinary by nature. While an *interaction* is a well-defined concept in physics the exercise is to make that concept useful in an engineering design context without compromising the rigor of the physics.

Having defined what an *interaction* is, it is now possible to reason about what an *interface* is again with the purpose of reducing ambiguity. RQ5 thus asks:

RQ5

How can an interface be defined and characterized, based on the understanding from the Interaction framework?

The answer to RQ5 is derived through logical deduction from the *Interaction Framework* while respecting the phenomena inherent in engineering design. The concept of an interface should therefore be compliant with notion of functional and physical domain views on products, while being unambiguous with regards to the definition of interaction.

The last research question, RQ6, investigates how to use the framework in practice from a method and tool perspective to create more complete interaction and interface specifications.

RQ6
How can the Interaction and Interface framework be applied in practice to support complete and consistent INTERACTION and INTERFACE specifications?

This question is answered through logical deduction from existing theory on systems design, architectural decomposition as well as on own experience.

See Table 2 for an overview of how the different papers relate to the different research questions.

Table 2 Overview of how the papers relate to the research questions

	Paper A	Paper B	Paper C	Paper D
RQ1	✓			
RQ2		✓		
RQ3		✓		
RQ4			✓	
RQ5			(✓)	✓
RQ6				✓

✓ = Means that the research question is *comprehensibly* addressed
(✓) = Means that the research question is *partially* addressed

Paper A and Paper B are thus clarifying the research and describing the phenomena from a theoretical and a practical perspective. Paper C and Paper D prescribe a new framework for reasoning about interactions and interfaces. RQ5 is addressed in both papers C and D in order to honor the need for 'stand-alone' papers, but the main contribution to RQ5 is written in Paper D.

1.5 Scope of the thesis

Theoretical scope (I.e. fields of research):

- *Engineering Design theory* and *Theory of Technical Systems*
- *Physics* is used as a foundation for the theoretical framework
- Focus is on characterizing the *object* itself, the product, and its interactions and interfaces
- Secondarily on the *development of the object*

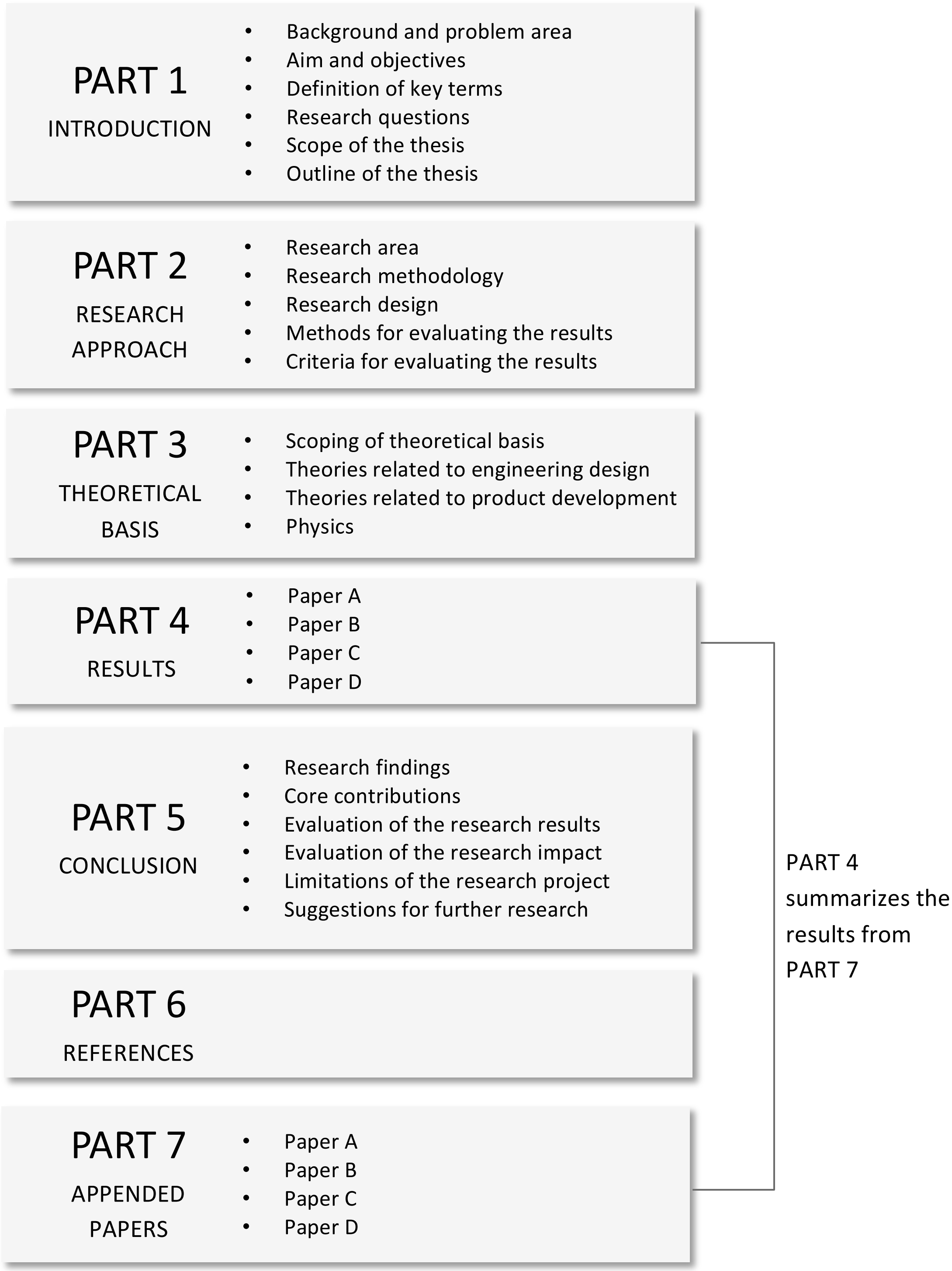
Practical scope (i.e. application):

- *Synthesis* and *analysis* of products
- *Complex, multi-technological, high-investment, high-risk projects*, where the consequence of failure is significant
- The *early architectural stages* of product development, *functional modeling*, *including interface embodiment*
- All technical disciplines relevant for engineering design (excluding nuclear engineering) are addressed

    - Software engineering is only included from the point, where SW commands are converted to electrical signals due to the very different nature of the discipline. Interfaces between modules of code are therefore not addressed
- Organizational aspects, such as responsibility and ownership of interactions and interfaces, are only treated lightly in this research. This will be subject for further research in order to improve the likelihood of adoption in industry
- The project does not address the proper 'vehicle' or software tool for operationalizing the theory. The project does also not treat aspects such as well as *version and revision history filing*. This will be subject for further research in order to improve the likelihood of adoption in industry

1.6 Outline of the thesis

The thesis is structured as follows:



2 Research approach

This section describes the overall research approach, which has been undertaken in this 3-year research project. The purpose is to expose how the research has been planned and conducted in order to provide credibility to the conclusions. The chapter is structured as follows; First, an overview of the Research area, and Research methodology. Then a description of the Research design, Methods for evaluating the results, and Criteria for evaluating the results.

This Industrial PhD project has been carried out at Radiometer Medical ApS in collaboration with the Technical University of Denmark (DTU), Department of Mechanical Engineering (DTU MEK) in Section of Engineering Design and Product Development.

2.1 Research area

This research project contributes to the area of *engineering design* however some of the theory, which is proposed, is derived from *physics* using a research method borrowed from *physics*. As such, this research project has ties into several fields of research. In order to create an overview, a diagram displaying the *areas of relevance and contribution* (ARC-diagram) has been created as suggested by (Blessing and Chakrabarti 2009). See Figure 3.

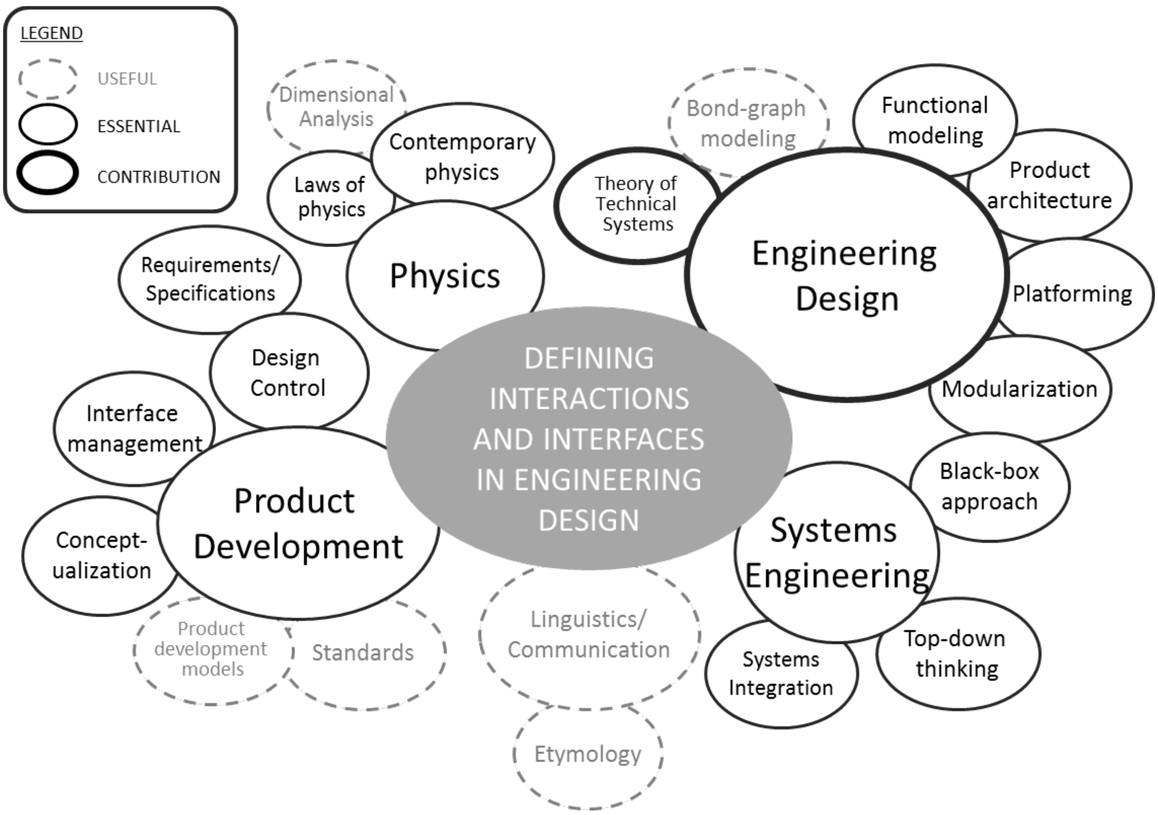


Figure 3 ARC-diagram showing an overview of relevant research fields and areas of contribution (Blessing and Chakrabarti 2009)

The main topic of the research project is stated in the greyed out bubble; "Defining interactions and interfaces in Engineering Design" especially in complex and multi-technological products. According to E

Crawley, Cameron, & Selva (2015) "*a complex system has many elements or entities that are highly interrelated, interconnected, or interwoven.*" Complex systems thus require systematic thinking in order to deal with the complexity. Another keyword is multi-technological that is defined by (Torry-Smith 2013) as a term, which "*relates to a 'product' [object]. A multi-technological product is a product comprising components developed by different engineering disciplines.*" Multi-technological products are thus boundary objects between multiple engineering disciplines meaning that the 'language' that is used to characterize the product may purposefully be defined in a way, which is understandable across multiple disciplines.

In general, theories and methods within *Design* focuses on either *the product* or *the process* or a combination of the two (Chakrabarti and Blessing 2014). Product *development* research thus distinguishes itself from *engineering design* research by developing theory, methods, and tools to support the process of developing a product (i.e. object), whereas engineering design focuses on describing the nature of the product itself (i.e. object) by articulating theory, models, concepts, and definitions, which support the characterization of the object (Andreasen 2011).

This research project primarily relates to the research area of *engineering design*, by deriving a theoretical framework concerning Interactions and Interfaces. Interaction is a basic concept of any physical system, including that of products. Interface, in this research project, is a term purely belonging to the engineering design field in terms of characterizing the structural connections between physical elements of the system, e.g. module interfaces.

The research project *contributes* to the *body of knowledge* of Engineering Design research by providing an unambiguous and complete, physics-based classification of interactions and interfaces, which is fundamental to characterizing the relations in and between technical systems, functional and physical elements in a product architecture. Physics is thus a foundation for this research although we do not contribute to this area.

The project touches upon *product development* and *systems engineering* related research by providing an 8-step architecting approach to defining interactions and interfaces during synthesis of complex, multi-technological systems.

Communication is considered as a useful field to this research project, because some of the causes of problem arising at the interfaces may be addressed by *communications theory*.

2.2 Research Methodology

From a methodology perspective, this research project can be categorized as *design research.* As of now, there is no single methodology within design research, which prescribes exactly how to structure a specific design research project because the diversity of topics is vast. For that reason, different approaches will be presented here, which are all more or less applicable to this research project. In section 2.3 Research design we will address how this particular project is designed and structured.

2.2.1 Problem- and Theory-based engineering design research approach

Jørgensen (1992) proposed a couple of stereotype work paradigms for how to perform research and development. As can be seen in Figure 4, research may have different starting points for reasoning. One approach is to analyze the problem base through empirical data and map out the phenomena related to the area of research resulting in a diagnosis. By synthesizing solutions to address the phenomena from the diagnosis the researcher may arrive at a new scientific discovery. The other approach takes its starting point from already existing theory and through the application of scientifically rigorous research methods the researcher is able to synthesize a new theory or model of understanding. By applying the theory or model in practice, the researcher is able to argue whether it is applicable to reality and useful. If so, the researcher may have arrived at a new scientific discovery.

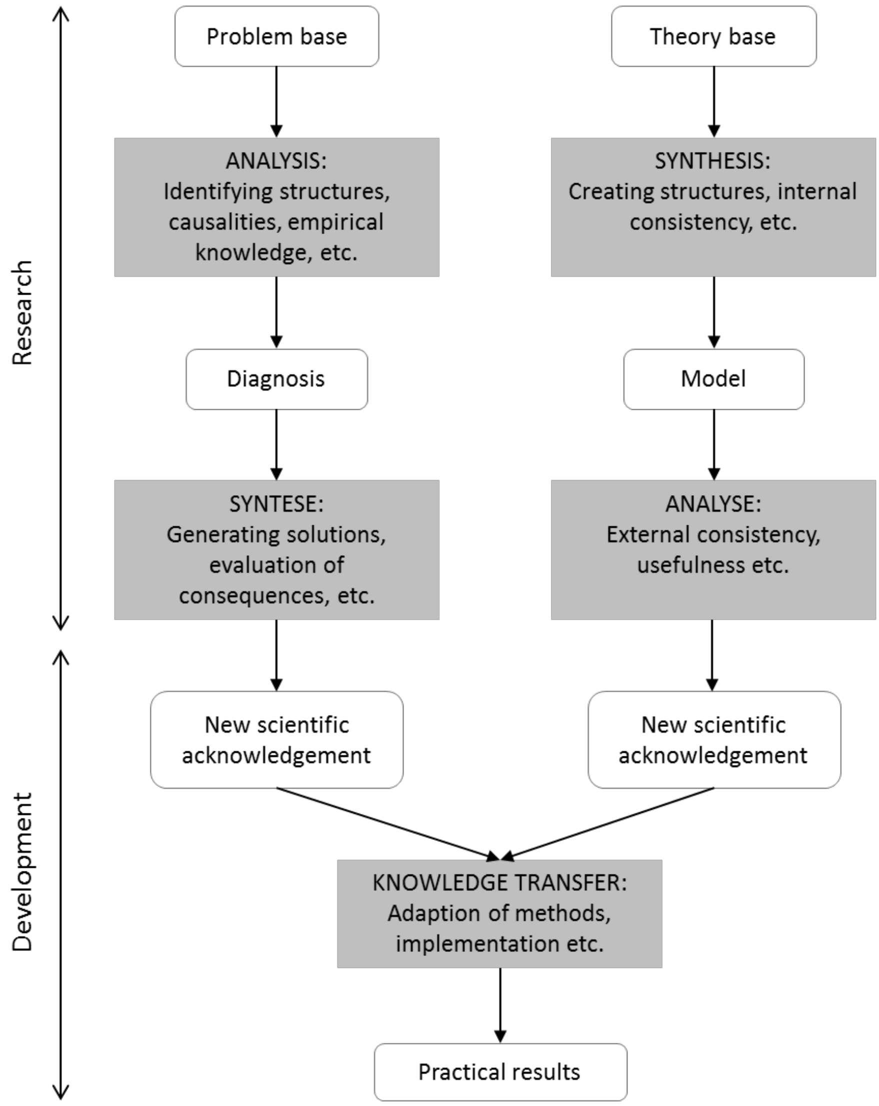


Figure 4 The Problem based, Theory based (PbTb) research approach - Foundational scientific work paradigms of research and development activities. Redrawn from (Jørgensen 1992)

The development part of this model describes a step where the research findings are transferred to practice, either through teaching, through collaboration between university and industry or through external consultancy.

This research methodology is rather sequential in its format, however in practice there is likely to be several loops between analysis and synthesis, and probably also between problem based and theory based research paradigms.

2.2.2 Design Research Methodology (DRM)

Blessing & Chakrabarti (2009) have proposed what they call the *Design Research Methodology* (DRM), which aims at raising the scientific rigor of design research. Design research is a very broad field and may encompass many different scientific disciplines that the research is often quite complex and diverse, making it difficult to follow a specific approach (Blessing and Chakrabarti 2009). DRM however outlines a structured approach to doing engineering design research based on several consecutive phases. See Figure 5.

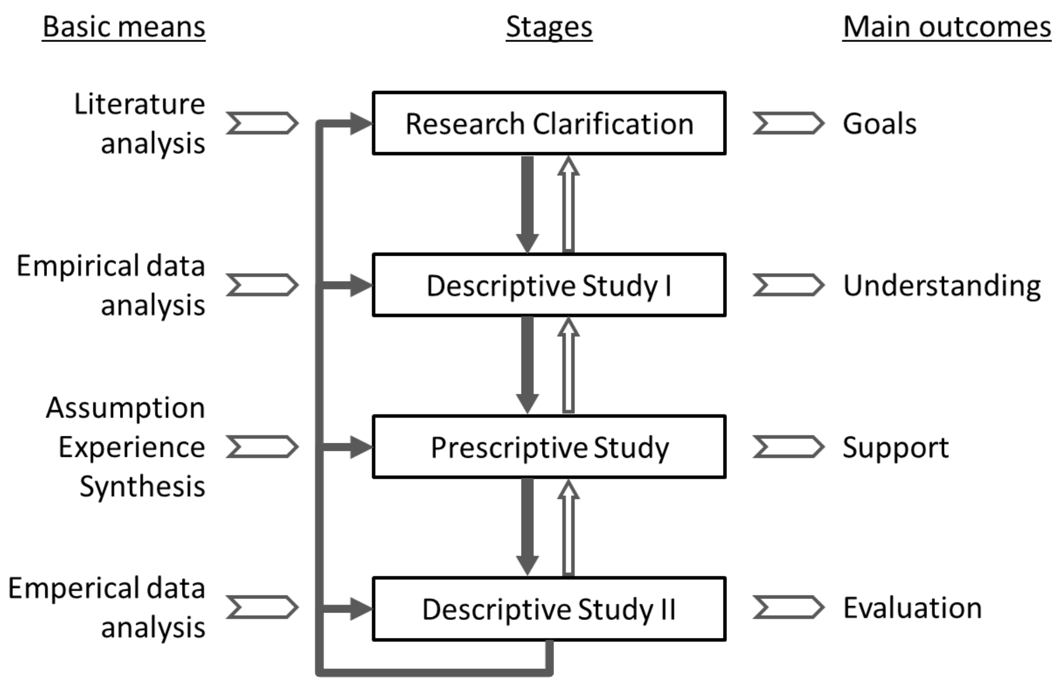


Figure 5 The DRM framework with the different stages and inputs and outputs from each stage. Redrawn from (Blessing and Chakrabarti 2009)

The research clarification stage is about identifying the goals and setting the scope of the research. This is followed by a descriptive study in which the phenomena are addressed and described. Based on this understanding, a prescriptive study is performed in which theory, methods or tools are developed to address the identified phenomena of interest. Finally, a descriptive study II is performed in order to assess how *good* the prescribed support addresses the phenomena.

The weight of the various stages may vary a great deal and given the time limitations of a PhD project (3 years in Denmark), certain stages may have to be scoped out. See Table 3.

Table 3 Types of design research projects and their main focus. The dotted box best represents this project. Redrawn from (Blessing and Chakrabarti 2009)

Research Clarification (RC)	Descriptive Study I (DS-I)	Prescriptive Study (PS)	Descriptive Study II (DS-II)
1. Review based →	Comprehensive		
2. Review based →	Comprehensive →	Initial	
3. Review based →	Review based →	Comprehensive →	Initial
4. Review based →	Review based →	Review based → Initial/ comprehensive ←	Comprehensive
5. Review based →	Comprehensive →	Comprehensive →	Initial
6. Review based →	Review based →	Comprehensive →	Comprehensive
7. Review based →	Comprehensive →	Comprehensive →	Comprehensive

2.3 Research design

The following section will describe how the research project was *actually* carried out in relation to the above mentioned research methodologies. The mental process of realization throughout the project has not followed a strictly stage-based sequential order. Rather, the gradual buildup of knowledge and insight continuously unlocked new ideas for investigating the phenomena and new ideas for types of support. The following overall cornerstones of realizations listed in chronological order thus confirms this:

- The literature review and informal interviews revealed that there is a need for a multi-disciplinary treatment of *interfaces*
- The design of *interfaces* is determined by the *interactions* that cross the *interface*
- The current classifications of *interaction* are not rigorous enough to claim that *all interactions* and *interfaces* can be captured unambiguously – we must start at the definition of *interaction* before commencing with defining *the interface*

Therefore, even though the initial starting point was to define an *interface*, we had to take a step back and look at *interactions* in order to arrive at a truly rigorous and multi-disciplinary interface concept. This realization was conceived around 1/3 into the project.

2.3.1 Research plan

The structure of the research project can be described in the following way:

- DRM constitutes the overall framework for this research, in particular the third type, see dotted line in Table 3

- The *research clarification* and *descriptive study I* have been performed based on a mixture of problem based and theory based approaches (see Figure 4) by both conducting interviews with domain experts as well as doing a literature review
- In terms of conducting the *comprehensive prescriptive study* this project has applied the *theory-based approach* from PbTb (see Figure 4), through a deductive approach from first principles physics coupled with the understanding of the phenomena in Engineering Design

In Figure 6, the grand overview of the project is presented as it was executed. It is structured around a chronological order with a timeline on top. Each stage of the project is listed with an indication of when the different research questions were investigated and answered. The various activities related to each stage are listed, including the associated methods that were applied. Finally, an overview of the publications is listed in the bottom.

The research project has more or less followed the structure as outlined in Table 3, dotted line (Blessing and Chakrabarti 2009), which was adopted 6 months into the project and finally planned half way into the project. Due to time limitations the scope of the project has been adjusted a bit compared to the original plan, e.g. a SW-based system budgeting/architecting tool and a requirements management tool have been scoped out and transferred to future research recommendations. Each stage of the research will now be described.

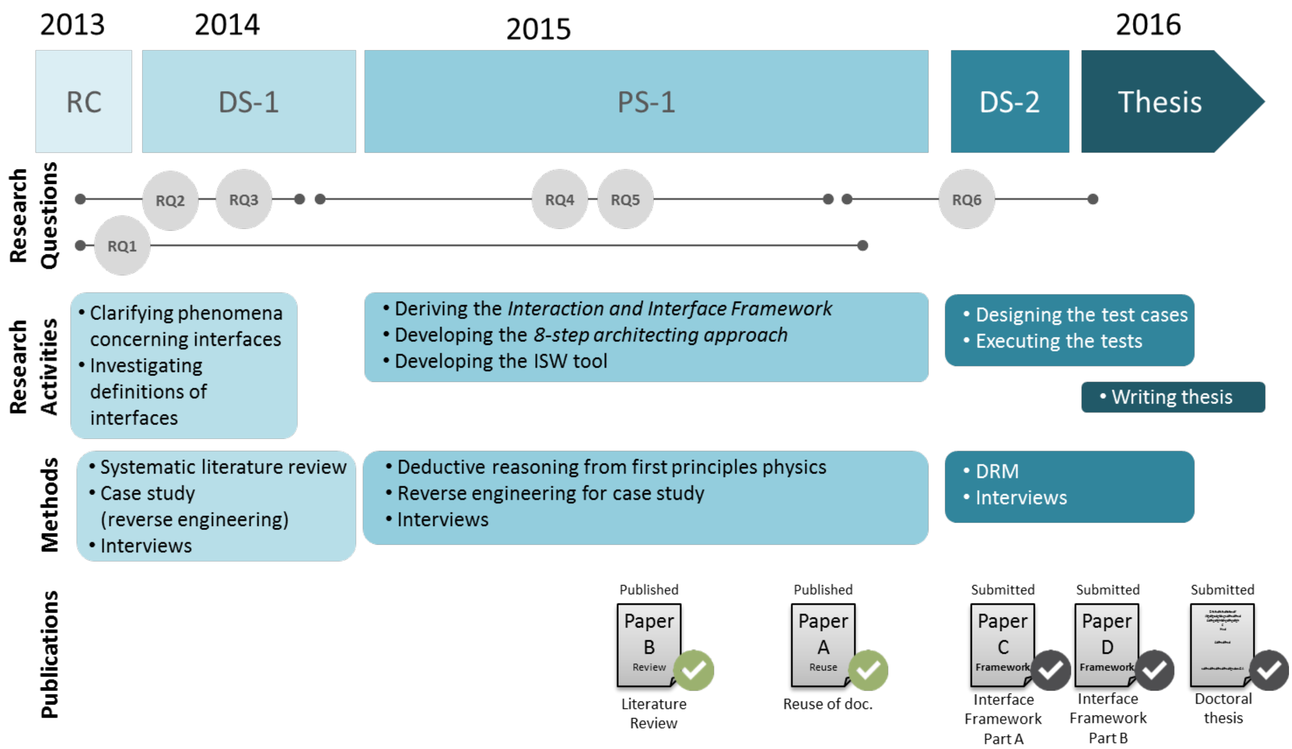


Figure 6 Grand overview of the three year project; research stages, associated research questions (RQ), research activities, methods applied and scientific papers. Green check marks mean 'Published', Grey check marks mean 'Submitted'

Research clarification (RC) and Descriptive Study I (DS-I)

The project was initiated with a light literature exploration to broaden the understanding of the phenomena and clarify the research field. Interfaces are evidently something, which is treated in many different fields of science; physiological science, communications theory, organizations theory, macro- and micro-political sciences, engineering etc. The list of analogies is endless, because of the fundamental nature of the phenomenon in question. However, an early decision was made to focus purely on *engineering sciences*, as well as *communication* although this has been treated very limited. Other early decisions were; multi-disciplinary products, complex systems with many components and interactions, only architectural part of product development. Also, a 2-year empirical case study was conducted in order to understand the role of interfaces in new medical device development. This supported the need for further research into the definition of an interface.

In the descriptive study I a systematic literature review on the definition of interfaces was then executed in order to reveal the state of the art concerning the perception and definition of an interface. A reverse engineering exercise of a solenoid valve was also performed in order to reason about some of the phenomena that might cause problems to arise at interfaces. The DS-I thus created an understanding of the phenomena concerning interfaces, and created the foundation for developing support.

Prescriptive study (PS) and Descriptive study II (DS-II)

The prescriptive study involved a comprehensive literature review into physics in order to obtain an understanding of the fundamental physics – first principles. A key reference in this phase has been Chabay & Sherwood (2011), who seek to unify the physical concepts to a few fundamental ones including simple mental models, in order to support the readers in reasoning freely across the physics branches. In order to make the physical concepts applicable to engineering design, we logically deducted a complete interaction classification from first principles and up a to a product scale. Based on this we reasoned out a definition and classification of an interface, which is compliant with all types of interactions across all technical disciplines, at any level of abstraction and concreteness, from both a functional and a physical modeling viewpoint. Also an 8-step architectural process and a calculation tool were developed in order to prescribe how the framework should be used in practice.

The descriptive study II was conducted in order to evaluate the applicability of the results. The framework was tested in five individual expert user tests. The tests were problem- and task-based mixed with a questionnaire, and interview session.

2.3.2 Research methods

One of the critique points of existing classifications of *interaction* is the fact that the classes overlap meaning that a certain physical phenomenon may be captured using several categories at once, which is ambiguous. In order to develop an *Interaction Framework*, which is *mutually exclusive* (i.e. no overlaps) and *collectively exhaustive* (i.e. no gaps) we have chosen to approach the derivation of the classification from a physics perspective. A core strength of this research project is therefore to apply a *first principle research method* known from physics in order to create a multi-disciplinary language suitable for engineering design, yet compliant with the laws of physics.

This approach will allow for a rigorous foundation for articulating *what* an *interface* is from a strictly multi-disciplinary perspective. We will therefore briefly describe what the method is about.

Deductive reasoning from first principles of physics

The idea of *first principles* dates back to Aristotle's work *Physics* (Irwin 1989), in which he writes about a proper method for arriving at the *first principles of natural things* (Mouzala 2012). The method has three steps; 1) inductive reasoning from a single perception of particular example into general features or universal characteristics, 2) analysis of a *whole* into its *parts*, 3) inductive reasoning considered as an advance from particulars to universals (Mouzala 2012).

The purpose of reasoning from first principles in this research project is not the same as prescribed by Aristotle, where you theoretically deduce the existence of *new* first principles e.g. hypothesizing a new elementary particle. Instead we take the starting point at the already established first principles of physics, the *laws of conservation* and *fundamental interactions*, and reason up to the *general characteristics* of an *interaction* from a product perspective with a clear and transparent line of reasoning. The purpose is therefore to contribute to the understanding of interactions as viewed and applied in the engineering design domain. See Figure 7.

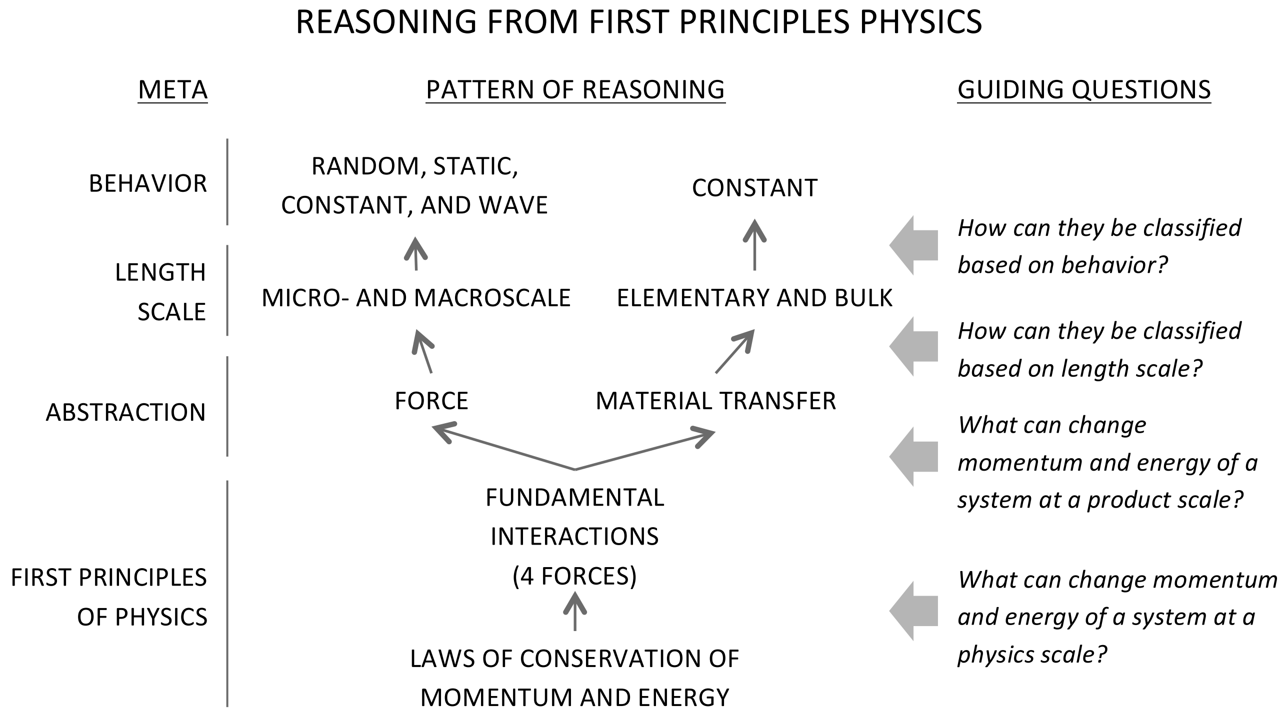


Figure 7 Meta-figure showing the reasoning pattern from first principle physics to classification of interaction in engineering design. The "guiding questions" illustrate the mental journey of exploration

The first principles approach stands in contrast to the typical way of deriving classifications in engineering design, which are typically empirical-based generalizations (i.e. case studies, reverse engineering, interviews etc.) The risk with a purely empirical approach is that the collection and analysis of data is influenced by the researcher's presence as well as interpretation and therefore may influence the *internal consistency* of the classification.

The powerful aspect about starting at the first principles of nature is that these are principles *knowable by nature* and not by *human perception* (Mouzala 2012). By starting from the very basic, established theories of understanding and moving up through means of inductive reasoning, without making assumption along the way, we are able to arrive at a classification not colored by human perception, which is m*utually exclusive* and *collectively exhaustive*. The uncertainty of using this approach is however related to the *usefulness* of the classification. See section 2.4 Methods for evaluating the results.

Case study method to investigate the role of interfaces in new medical device development

In order to investigate the phenomena concerning interactions and interface in new product family development, a 2-year empirical case study was undertaken of two families of arterial blood gas samplers (Yin 2013). The case study involved codifying and reviewing large amounts of product documentation, interviewing domain experts, and modeling using well-known modeling methods such as Design Structure Matrices (DSM) (Steward 1981; Eppinger and Browning 2012).

It has been a core objective of this case study not to affect the situation, which is being observed and analyzed while carrying out the case study. Because the case study mainly relied on historical data, the researchers influence is limited.

Overview of methods

Various methods have been applied throughout the project to answer the six research questions. See Table 4 for a complete overview.

Table 4 Overview of the different methods applied in the project. Columns represent the methods whereas research questions are listed in the rows. R = Researcher, P = Participant. RR / PP = an extra high effort

	Methods						
	Case study	**Interview**	**Reverse engineering**	**Document analysis**	**Literature review**	**Logic/ deduction**	**Prototyping**
Research questions	arterial blood gas samplers	Domain experts from different disciplines	Solenoid valve	Design control documents	Systematic literature review (Levy and Ellis 2006)	From *first principle* physics	Paper prototype of tool
RQ1	RR / P	R / PP		RR			
RQ2					RR		
RQ3		R / PP	RR				
RQ4						RR	
RQ5						RR	
RQ6							R / P

The first four methods are *descriptive* in character, which corresponds with the three descriptive research questions 1, 2 and 3. The last two methods have a *prescriptive* character and are applied in developing the support, thus answering research questions 4, 5 and 6. For explanations on the above mentioned methods we refer to (Blessing and Chakrabarti 2009).

The case products, analyzed using reverse engineering are displayed below, see Figure 8.

Figure 8 Left: Product families of *arterial blood gas samplers* used for Paper A. Right: A *solenoid* used as a multi-technological case example for understanding the phenomena in question, used for Paper B

The arterial blood gas samplers to the left in Figure 8 were chosen as case product because a lot of material could be analyzed in order to understand the role of interactions and interfaces in product family development in the medical device domain. The solenoid case product to the right was chosen because of its multi-technological nature, and because it was interfacing with a greater system, thus making it useful for reasoning about the issues concerning the definition and perception of interfaces in practice.

2.3.3 Research activities

During the course of 3 years, I have exchanged knowledge about my research with peers from all over the world. The purpose has been to seek inspiration, challenge my ideas and beliefs, and create an international network of frontiers within this particular research topic. The following list outlines the research activities:

Dissemination of knowledge

Throughout the project I have spent a lot of time on sharing my knowledge and get inspired through presentations and dialogs with others. The inputs and feedback have been very useful for shaping and scoping the project along the way. The following activities represent the achievements in this regard:

- Several presentations at host company Radiometer Medical ApS, 2013-2016
- Guest lecture in the class; *Product Platform and Product Family Design: From Strategy to Implementation,* MIT, Cambridge, MA, USA (see Figure 9), July 2015
- ASME proceedings, IDETC/CIE DTM presenting Paper A, August 2015
- 2nd Spring school on Systems Engineering, TU Munich, May 2014
- Summer School on Engineering Design Research Methodology, June/July 2014
- External company presentation at FOSS, May 2015
- Knowledge exchange visit to companies Beckman Coulter and AB Sciex, USA, October 2013
- Book project on Conceptual Design published 2015 (Andreasen et al. 2015), 2013-2015
- Design and supervision of four individual MSc thesis projects, 2013-2015

Figure 9 Guest lecture in *Product Platforms and Product Family Design – From Strategy to Implementation*, MIT, Cambridge, MA, USA, July 2015

Courses

The various courses have provided insight into research methodologies for ensuring rigorous research results, insight into closely related fields of research, as well as the chance to discuss my research with fellow PhD students and faculty from various research institutions around the globe; Denmark, Sweden, Germany, UK, Portugal, Italy, India, USA etc.

- Business course for Industrial PhD students, DTU, 2013 (7.5 ECTS)
- Design Research Terms and Methods for PhD Students, DTU Mechanical Engineering, 2013 (5 ECTS)
- 3-day Master class in Systems Engineering, ITOS, 2013 (No credits)
- Summer School on Engineering Design Research, Italy/Germany, 2014 (5 ECTS)
- 2nd Spring school on Systems Engineering, TU Munich, 2014 (3 ECTS)
- Product Platform and Product Family Design: From Strategy to Implementation, MIT, Cambridge, MA, USA, 2014 (5 ECTS)
- Systems Engineering, Architecture, and Lifecycle Design: Principles, Models, Tools, and Applications, MIT, Cambridge, USA, 2015 (6 ECTS)

Conference attendance

A highly profiled conference on *Design Theory and Methodology (DTM*) was attended in August 2015, where I provided a presentation of Paper A and networked with other researchers. See Figure 10. The paper was recommended for journal publication and honors by double blinded peer reviewers.

- DTM, 2015 ASME proceedings IDETC/CIE DTM, Boston, MA, USA

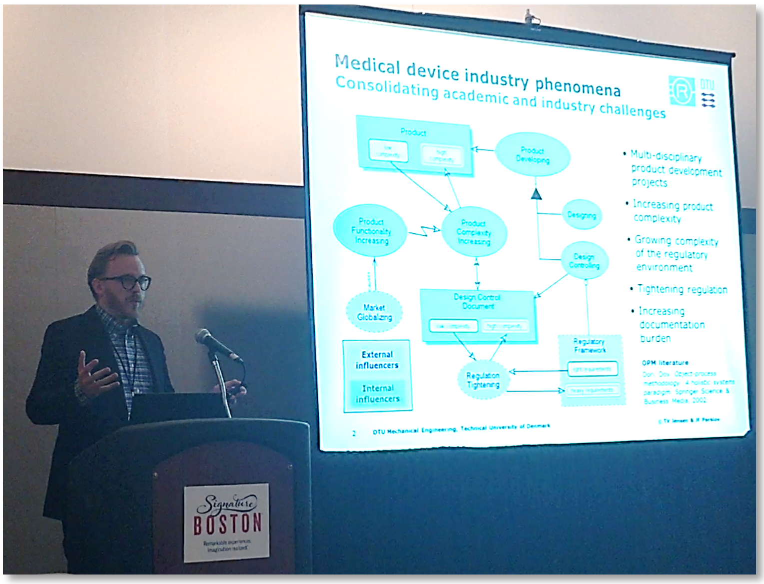


Figure 10 Presentation of Paper A at IDETC/CIE DTM-5: Design of Complex Systems and Product Architecture, ASME proceedings, Boston, MA, USA, August 2015

A 1.5 months research stay was held at the Astro- and Aeronautics Department at MIT, Cambridge, MA, USA, courtesy of Prof. Olivier de Weck. This research stay allowed for an informal exchange of my research with world-class researchers at the MIT Engineering Systems division.

2.3.4 Role of the researcher

As an Industrial PhD student I have been considered as an integral part of the organization at Radiometer Medical ApS. I have been actively engaged with the development activities providing input and reflections. Despite my active involvement in practice, the prescriptive research has not been influenced by it because of the theory-based approach to conducting the prescriptive study. My role as a researcher/employee does therefore not serve as a basis for concern, i.e. no bias.

2.4 Methods for evaluating the results

Evaluation of *Design research* is challenged by the stochastic nature of the design activity meaning that new methods and tools are not necessarily a guarantee of a better result because the conditions change (Buur 1990). Because of the vast amount of influencing factors it may not be possible to replicate an experiment in real-life and thus prove the validity of the research contributions (Buur 1990).

This section will therefore introduce various methods of *verifying* and *validating (V&V)* the research results. The distinction between *verification* and *validation* of this research can be articulated similar to a popular saying:

- Verification: Are we conducting the *research right*?
- Validation: Are we conducting the *right research*?

Because this research project has elements of both qualitative *empirical* research (RC, DS-I, DS-II) as well as qualitative *deductive* theory-based research (PS), the distinction between verification and validation becomes highly relevant when discussing the various methods for evaluating the results. As there is no single method for evaluating design research we choose to present three applicable approaches.

2.4.1 V&V principles from mechatronics research

Buur (1990) suggests in his doctoral thesis two types of verification/validation of the research:

- *"Logical verification*
    - *Consistency: there is no internal conflicts between individual elements (e.g. axioms) of the theory,*
    - *Completeness: all relevant phenomena observed previously can be explained or rejected by the theory (i.e. observations from literature, industrial experience etc.),*
    - *Well established and successful theories and methods are in agreement with the theory*
- *Verification by acceptance* [ed. this is understood as *validation* in accordance with the above explanation]
    - *Statements of the theory (axioms, theorems) are acceptable to experienced designers*
    - *Models and methods derived from the theory are acceptable to experienced designers"*

(Buur 1990)

According to Andreasen, (2011) the essential goal of design theory is for the theory to lead to "*productive designing through the created mindset of the designer and the models, methods, and tools"* (Andreasen 2011). However, it does not make much sense to test the *external validity* if the *internal consistency* of the theory is poor.

This approach from Buur (1990) may therefore be a suitable fit to this project because it touches upon both aspects; rigor of the conducted research (internal, verification) and applicability of the research (external, validation).

2.4.2 V&V principles from design research

Blessing & Chakrabarti (2009) propose three central criteria for evaluating design research, which can be categorized as follows:

- Application evaluation
    - *Usability*: Whether the developed support can be used, and with what ease
    - *Applicability*: Whether the developed support has the direct effect on the phenomenon in question
- Success evaluation
    - *Usefulness*: Whether the developed support affects the measurable success criteria in a real-world project

(Blessing and Chakrabarti 2009)

The three criteria (i.e. usability, applicability, and usefulness) are all evaluations of the output from the prescriptive support. This approach does therefore assume, that if the end result is acceptable (i.e. validated), the internal construct of the method must also be 'correct' (i.e. verified), but this may not always be the case.

2.4.3 The Validation Square

(Pedersen et al. 2000) present a systematic approach to validate design methods called "The Validation Square". This framework builds on the authors' understanding of knowledge validation as being linked to *contextual usefulness*. They claim that *total objectivity* does not exist and therefore adopts a *relativistic view* on scientific knowledge. They define that "*knowledge validation becomes a process of building confidence in its usefulness with respect to a purpose*" (Pedersen et al. 2000).

Usefulness is further characterized as the *effectiveness* of the design method (i.e. whether the method provides the intended result/output) and the *efficiency* of the design method (i.e. whether the result/output has an acceptable performance and has been derived with less cost and/or in less time) (Pedersen et al. 2000).

The whole idea of the Validation Square is therefore to ensure *internal consistency* and *external relevance* from both a *theoretical* and an *empirical* point of view of the design method (i.e. prescriptive support). See Figure 11.

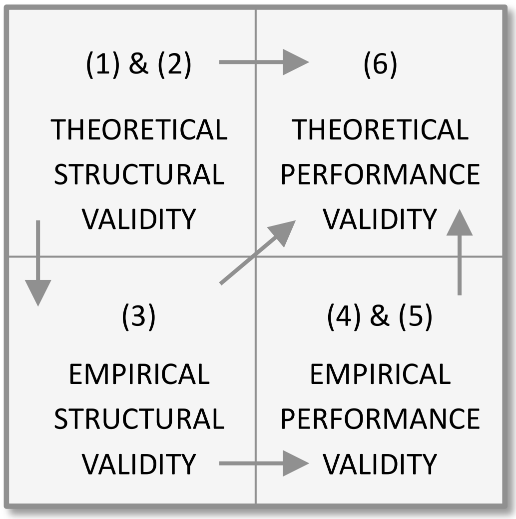


Figure 11 The Validation Square. Numbers 1-3 represents three aspects that characterizes *effectiveness* and 4-5 characterizes efficiency (Pedersen et al. 2000)

Effectiveness (*structural validity/internal consistency)*:

1) *"Accepting the individual constructs constituting the method*
2) *Accepting the internal consistency of the way the constructs are put together in the method, and*
3) *Accepting the appropriateness of the example problems that will be used to verify the performance of the method"*

(Pedersen et al. 2000)

Efficiency (performance validity/external relevance):

4) Accepting that the outcome of the method is useful with respect to the initial purpose for some chosen example problem(s)
5) Accepting that the achieved usefulness is linked to applying the method, and
6) Accepting that the usefulness of the method is beyond the case studies

(Pedersen et al. 2000)

2.5 Criteria for evaluating the research impact

In this section I present several criteria for evaluating the research impact. These criteria reflect the overall goals of the research project and visually illustrate my understanding of the phenomena, see Figure 12. The purpose of creating this *impact model* is to ensure a rigorous line of reasoning from the overall research objectives down to the key factors that we direct our support to. Thus the impact model reflects the intended impact of the prescriptive support. The model has been updated continuously following the gradual acquisition of knowledge about the phenomena.

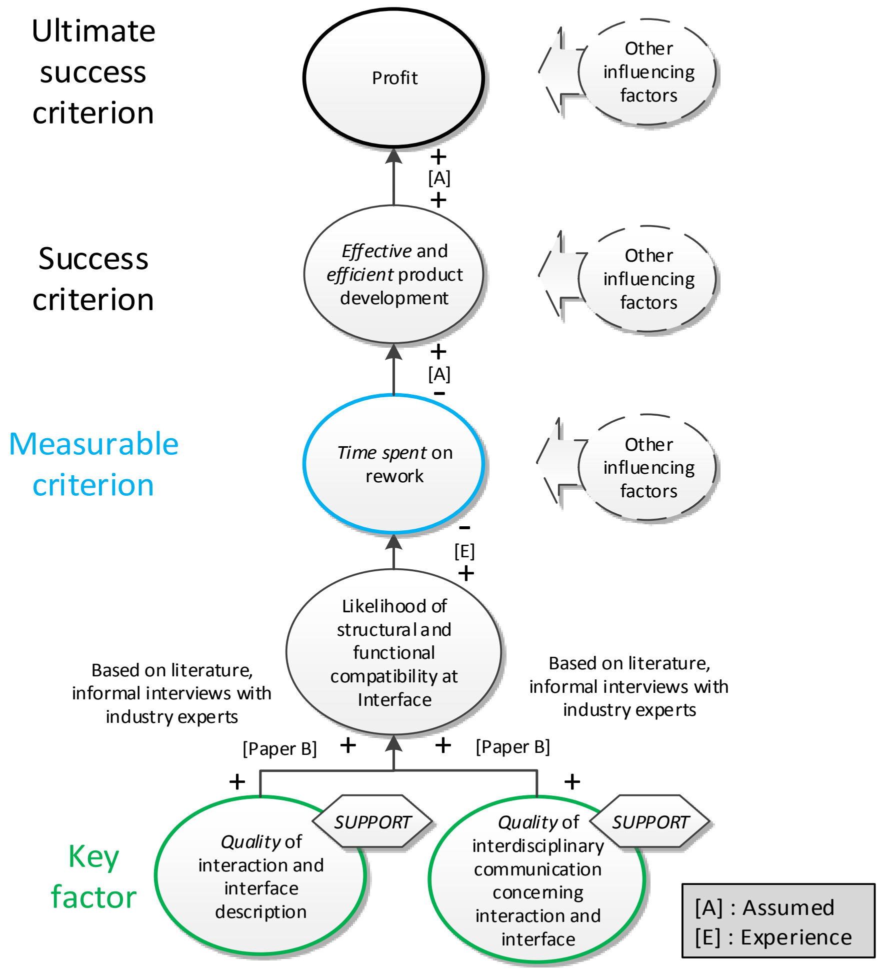


Figure 12 High-level *impact model* based on Blessing & Chakrabarti (2009). The key factors are considered as the most useful influencing factors to support, in order to achieve the desired goal of the research, the success criterion

The model contains nodes, which are *influencing factors*. They are connected by *causal links,* which describe the cause and effect relationships between them. The plusses and minuses determines the value of the attribute in the element, e.g. "+ *time spent* on rework" means that a large amount of time spent will lead to some other factor depending on the direction of the arrow.

High-level impact model

The *ultimate success criterion* for the prescriptive support is to lead to a profit gain for the companies applying prescriptive support from this research. It is not possible to directly correlate the influence of this research project on the overall profit of the operating company. For that reason the model outlines upstream factors, which may lead to the ultimate success criterion.

In order to achieve a profit gain, the prescriptive support must lead to *effective and efficient product development,* which is the *success criterion*. Here, *effective product development* is about producing the intended results whereas *efficient product development* is about producing the same intended result with a minimum use of time and resources. It is however not possible within the time frame of this research project to quantify this effect. Therefore, the *measurable criterion* looks at the time-spent on rework as a result of incompatibilities arising at interfaces.

The two key factors leading to incompatibilities at interfaces are identified as *the quality of interaction and interface description* as well as *the quality of interdisciplinary communication concerning interaction and interface.* Whereas the former key factor characterizes the object, the description of interaction and interface, the latter has to do with the language or communication concerning interactions and interfaces across multiple engineering disciplines.

While the high-level impact model is useful for illustrating the practical purpose of this project, is has not been possible within the time-frame of this project to apply the prescriptive support, the *Interaction and Interface Framework*, in a real-world project. We therefore redefine the impact model by looking at the upstream factors that lead to the high-level-impact model.

Low-level impact model – What is 'goodness'?

The following low-level impact model thus elaborates on the causes leading to the key factors in the high-level impact model. These key factors are converted into the success criterion in this low-level impact model. See Figure 13.

The question which is essentially addressed in this low-level impact model is; *what is 'goodness' concerning interaction and interface?* The ability to articulate *'goodness'* in this area therefore relies on an understanding of the underlying phenomena.

As stated earlier, we distinguish between the *quality of the description itself*, which is read and interpreted by engineers of various backgrounds, and the *quality of the interdisciplinary communication* concerning interactions and interfaces.

The model in Figure 13 shows that the causes of ***quality interaction and interface description*** have to do with the number of interactions, which are captured at an interface given the knowledge or uncertainty inherent in the system at a certain point in time. The key factor causing this number is the ***completeness of interaction and interface descriptions***. This is supported by providing a complete *Interaction classification* capable of capturing *all* physical interaction phenomena and can therefore be used as a checklist. See Paper C.

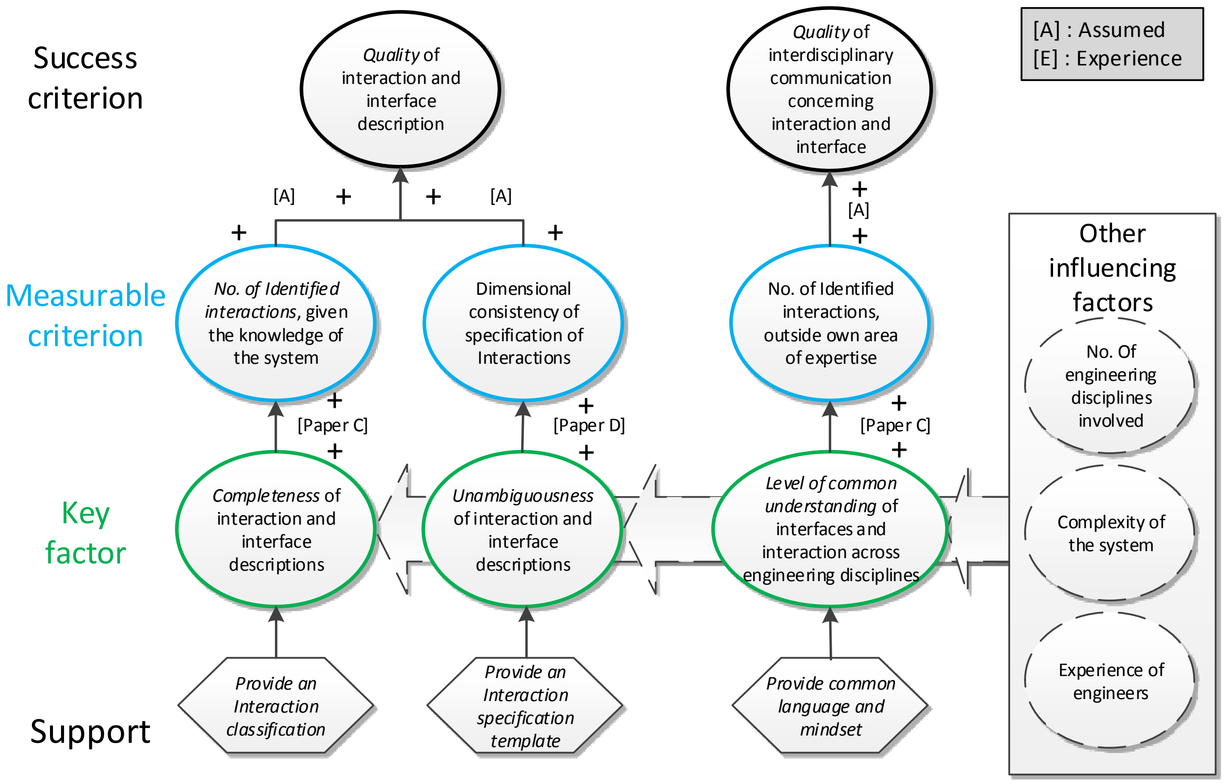


Figure 13 Low-level impact model outlining the criteria for the prescriptive support

The other factor has to do with specification consistency of the interactions, which can be evaluated using *dimensional analysis* (Mahajan 2014). The consistency is caused by the *unambiguousness of interaction and interface descriptions*. This is supported by providing an *Interaction Specification Template* as well as a tool called an *Interaction Specification Wheel (ISW)*. See Paper D.

The other success criterion of the low-level impact model is about the *quality of interdisciplinary communication concerning interaction and interface.* This aspect has to do with engineers understanding each other or speaking 'the same language'. The measurable criterion that leads to the success criterion is identified as the *number of interactions that an engineer identifies outside his or her area of expertise*. This criterion is therefore meant as a proxy of how well the engineer is able to reason freely about other interaction that falls outside his or her area of expertise.

The key factor, which leads to the measurable criterion, is *the level of common understanding of interaction and interfaces across different engineering disciplines*. To support the key factor, this research project provides a *common language* and mindset for speaking and reasoning about interactions and interfaces independent on technical discipline.

All of these key factors are assumed to be under influence of certain factors related to the engineers developing the product and to the product itself such as the number of engineering disciplines involved in a development project, the complexity of the system and the experience of the engineers. Complexity can further be decomposed into the actual, objective complexity of the product, and the perception of complexity that an engineer might impose on a product – the level of *complicatedness*. These factors have therefore also been taken into account in the test protocol for evaluating the framework.

3 Theoretical basis

The following Part 3 introduces the theoretical basis underlying this research. The purpose is to outline various theories and explain how they support this research project as well as position this research according to the existing knowledge base. Part 3 will begin by scoping the theoretical basis followed by descriptions of theories related to engineering design, product development, and physics.

3.1 Scoping of theoretical basis

The aim of this research project has been to derive a theoretical framework, which is universal in describing all physical interactions using a few basic concepts and a simple mental model. Through a transparent and rigorous approach for defining the terms and concepts, it is the intention that any 'school' within the engineering design research community may adopt this theory in their conceptual theoretical framework. The rather bold aim of *universality* has been pursued using a first principles approach from fundamental physics in order to derive a *mutually exclusive* and *collectively exhaustive* classification of interactions for use in engineering design.

The contribution of this research therefore falls within the field of *engineering design* meaning that we contribute to *the meaning and understanding of the nature of products as design objects*, more specifically *the nature of interactions and interfaces*. The contributions also extend into the field of product development, in terms of addressing *how to develop the design object and systematically define interactions and interfaces top-down.*

Therefore, this chapter will present the theoretical background for this research covering three main scientific fields:

- Engineering Design
- Product Development
- Physics

The selected theories within each field provide a fundamental understanding and language for speaking about products and product development. The engineering design and product development section will primarily contain theories from the 'Copenhagen school' in order to maintain a consistent line of reasoning through the text. The physics section will address the field of contemporary physics and why this movement within physics is an enabler for this research.

3.1.1 A model of design research

Duffy and Andreasen (1995) have proposed a research approach for design science that explains how models of phenomena are based on "reality" of designing as well as on theories, which explains these phenomena.

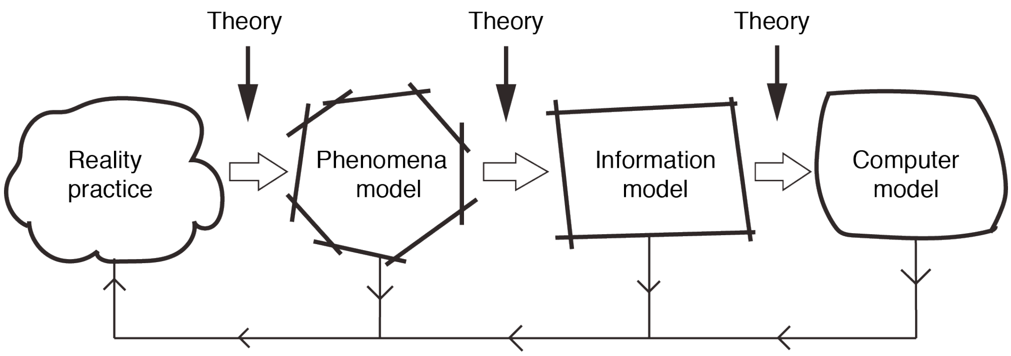


Figure 14 In design science models are derived from practice and developed for practice to enable productive designing. Redrawn from (Andreasen 2011)

Unlike natural sciences (e.g. physics) where the purpose of a model is to improve *predictiveness* of the behavior of a certain phenomenon, the purpose of models in design science is to *affect* the behavior of designing as a phenomenon, into producing better and more consistent results (Andreasen 2011). The nature of designing is thus more goal-oriented than natural sciences. The practice of designing is complex due to the complex pattern of influencing factors.

This model by Duffy and Andreasen (1995) provides a mental framework for understanding the positioning of this project in a research context. The key point here is that whereas the descriptive part of this project relies on empirical evidence from practice to derive phenomena models, the prescriptive part is mainly logically derived from physics (i.e. natural sciences). The prescriptive contributions from this project add to both *design theory* and *phenomena models*.

The following questions will be answered for each theory as a general format:

- What is the theory about?
- How does this theory support this research?

3.2 Theories related to Engineering Design

Engineering design research relates to *the artifact* being developed - the product. It has to do with the nature of the product and its relations to its environment. As such, the main contribution of this research, the *Interaction and Interface Framework*, will contribute to this particular area of research.

3.2.1 Systems theory

Systems theory is a meta theory, which provides a conceptual basis for other theories (Mortensen 1999). Because of its meta-level nature, it is not very useful in describing specific phenomena related to engineering design. Much research within engineering design builds on systems theory and further characterizes the generic concepts into useful terms with specific meaning.

The following model illustrates the general notion of systems theory. See Figure 15.

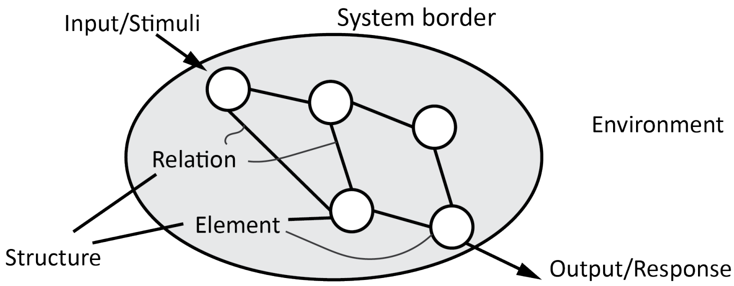


Figure 15 A basic model of systems theory. The model is recursive meaning that each element (i.e. node) in the system may in itself be a system with elements and relations. Redrawn from (Hubka and Eder 1988)

Hubka & Eder (1988) define a system as "a finite set of *elements* collected to form a whole under certain well-defined rules, whereby certain definite *relationships* exist between the elements, and to its environment." Any system has a system boundary across which inputs and outputs take place. A core concept of systems theory is the notion of *recursivity*, meaning that an element of a system may in itself be considered as a system, with its own elements and relations. A system may therefore be divided into partial systems called sub-systems. The reverse way is also possible, i.e. that a particular system, is an element of a greater system. The notion of *system of systems* is based on this principle (Haskins et al. 2006).

Any system has structure, i.e. the organization of elements and their relations, and behavior, i.e. the output/response relative to the input/stimuli.

Systems can be classified as suggested by Hubka and Eder (1988). They claim that systems can be classified as either natural or artificial system. See Figure 16.

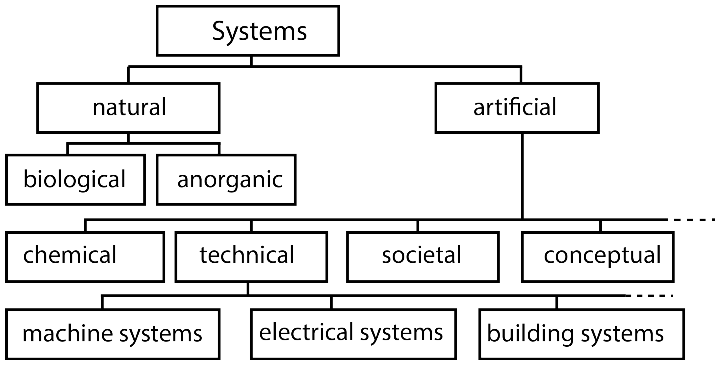


Figure 16 Classification of systems. A technical system is an artificial system conceived through human intervention. Redrawn from (Hubka and Eder 1988)

A key difference between artificial systems and natural systems is that artificial systems are conceived through human intervention.

How does this research project relate to Systems theory?

Systems theory is an essential foundation for this research project, in that it provides a framework for modeling reality with a few generic concepts such as *system boundary*, *elements* and *relations*. In this thesis, the use of systems theory for describing the framework is deliberate because of the objective to create a universal framework applicable to all schools of engineering design. Technical systems are further seen as a foundation for reasoning about product. The following theories apply systems theory as an underlying framework for describing products and the design activity and provide meaning to the systems terms.

3.2.2 Transformation Systems

In order to understand the purpose of technical systems we must look at the broader picture – *transformation systems*. Transformation systems encompass all *operators* and *processes* necessary to transform an input to an output. The transformed inputs are called *operands*, which change state during the transformation process. This process is facilitated by the effect of a number of different *operators* interacting with each other, e.g. operators being technical, human, information, and management systems etc. See Figure 17.

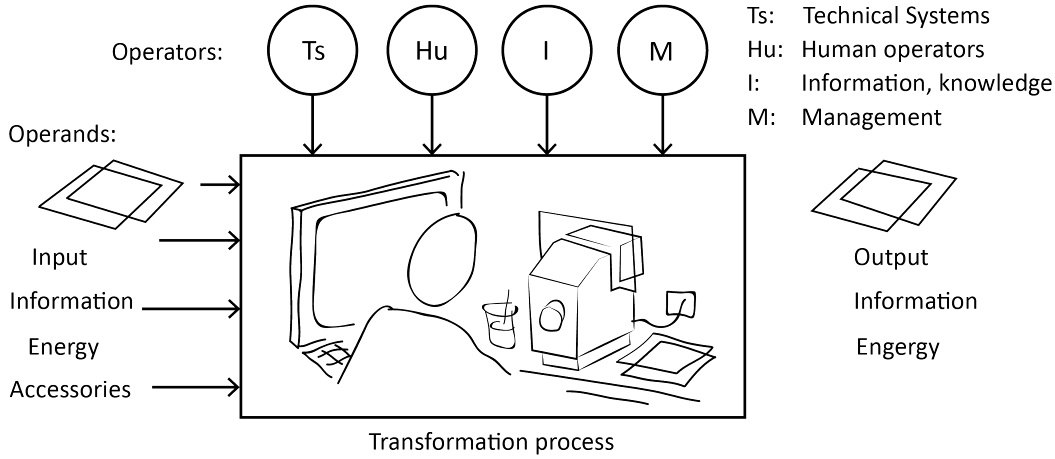


Figure 17 Overall transformation system. Transforms operands, i.e. materials, energy, information, bio. objects, from an input state to an output state. We are interested in the technical system. Redrawn from (Hubka and Eder 1988)

The notion of a transformation process may thus be considered as a 'black box' system where only the inputs and outputs states are defined.

In order to narrow the scope, Hubka & Eder (1988) further characterizes the *technical process*, which is an instantiation of a transformation process, where a technical system is an active operator. The technical process involves the collective effort of a technical system, human system, and active environment to transform the state of an input operand to an output operand.

3.2.3 Operands

An operand is a general term for any object which is changed (Hubka and Eder 1988). The change is aided by effects exerted by humans and/or technical systems, e.g. a material can change shape through the effect of a human hand (i.e. human system) molding the material or by being extruded through a meat grinder (i.e. technical system). An operand can be classified according to the following classes (Hubka and Eder 1988):

"

a) *Biological objects. Such objects consist of living individuals or groups of human, animal or plant life-forms. Within the technical process applied to these biological objects, their state (e. g. sick-+ healthy) or their location can be transformed.*

b) *Materials. Within the technical process, the transformation affects their basic properties, structure, form, dimensions, location, etc.*
c) *Energy. Within the technical process, either various types of energy (or energy carriers) are transformed into others, or their parameters are changed (e. g. p1 -+ p2, t1 -+ t2), or both kinds of change occur simultaneously.*
d) *Information. This set comprises commands (requests, desires, rules, normative statements), and data (verbal, graphical and symbolic/numerical). The transformation concerns the form, quality, quantity and location of information within the information carriers.* "

(Hubka and Eder 1988)

The technical and human systems are important *operators* of a technical process.

3.2.4 Technical System

A technical system refers to any physical artifact, which has been conceived through human intervention. A technical system is an artificial system where physical phenomena are exploited and arranged in a way that transforms an input to a desired output. A technical system has *structure* in terms of a physical manifestation consisting of components and *behavior* meaning the physical outputs as a function of its inputs. Hubka & Eder (1988) state in their proposition 7.2 (page 237) that "*the behavior of a technical system is determined by the structure of that system.*" And further in proposition 7.4 (page 237) that "*the observed behavior does not uniquely determine the structure that caused it. The same behavior can be realized by a number of different structures.*" Thus, there is a causal relationship from the structural characteristics of a technical system to the behavioral properties and *not* the other way around.

Synthesizing a technical system ideally progresses from defining the intended behavior of the system (i.e. the intended effect on an operand during a technical transformation process) to realizing the technical system by finding suitable physical structures that exhibit the intended behavior. A technical system can purposefully be modeled as a *black box* in which only the inputs and outputs are known, see Figure 18.

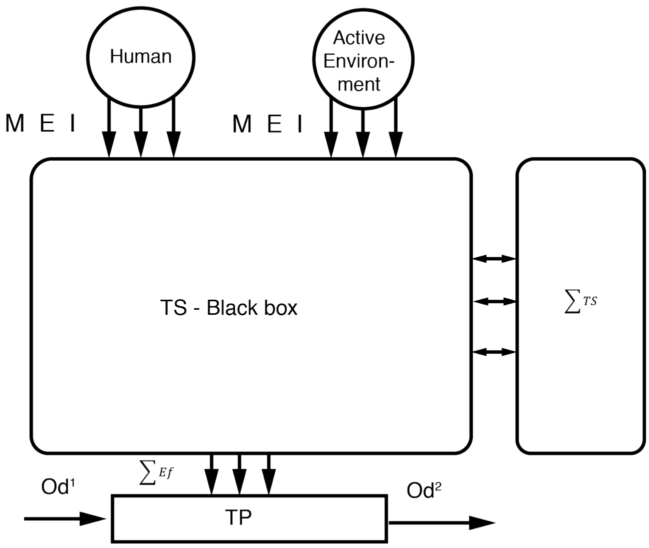


Figure 18 Model of Technical System (TS) as a black box with relations to the technical process, other TSs Human systems, Environment. Redrawn from (Hubka and Eder 1988)

A technical system may be acted upon by a human system, an active environment, and other technical systems ($\sum TS$), as can be seen in Figure 18. The technical system then exerts its effect in a technical process to support the transformation of an operand from one state to another.

The *mode of action* of a technical system is a description of "the way in which inputs of a technical system are converted into its effects (its outputs)" (Hubka and Eder 1988). The mode of action of a technical system thus describes 'how a technical system works' but does not necessarily describe how it eventually behaves once activated in its intended environment.

3.2.5 Couplings

According to proposition 7.8 (page 237), "*the behavior of a technical system depends not only on the behaviors of the elements, but also on the coupling relationships, between these elements*" (Hubka and Eder 1988). This phenomenon is also identified as *emerging properties* by (Crawley et al. 2004; Weck et al. 2011; Crawley et al. 2015). A coupling occurs whenever an output from one element or system, is input to another element or system.

Hubka & Eder (1988) state that there are different *couplings* between system elements; mechanical, electrical, chemical, magnetic, time or space couplings or any useful combination of these (Hubka and Eder 1988). These are listed as typical examples of couplings in machine systems. A more generalized concept of couplings is that they are either material, energy, or information (commands or data) (Hubka and Eder 1988).

Hubka & Eder (1988) add that it is *outside the scope of their book*, to contribute with a complete classification of couplings.

How does this research project relate to Theory of Technical Systems?

The TTS provides a language and conceptual framework for understanding and speaking about products as technical systems. This research project thus also considers products as technical systems, and further elaborates on the concept of *couplings* as defined in TTS, however using a different terminology.

3.2.6 Theory of Domains

The Theory of domains (ToD) was conceived by Mogens Myrup Andreasen in his doctoral thesis in 1980 (Andreasen 1980). The theory has undergone some changes over the years, however the most recent version will be presented in this chapter (Andreasen 2011).

The core contribution of ToD is the notion that products, or technical systems, can be viewed from three different viewpoints; the *activity domain*, the *organ domain*, and the *part domain*. See Figure 19.

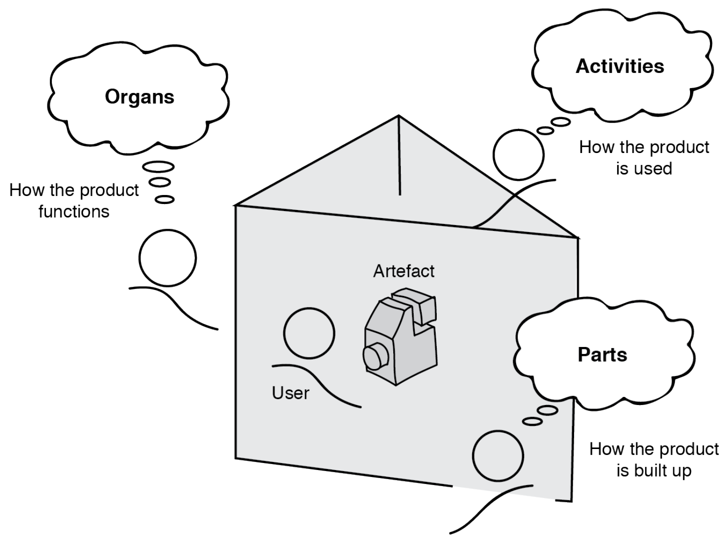


Figure 19 A popular depiction of the three domains from which a product can be viewed. Redrawn from (Andreasen 2011)

The *activity domain* articulates *how the product is used*, by modeling the activities that a product takes part in, and the state changes it undergoes during the technical process. At each activity step in the technical process, the product may be activated and interacted with by its user (the human system) to deliver a certain effect that will transform the state of one or more operands into an output state.

The *organ domain* articulates *how the product works,* i.e. the *mode of action* of the product. The organ domain consists of organs and their relations, which are functional interactions of the kind material, energy, information, or biological objects. Organs are defined as "a system element of a product […] is characterized by its function and mode of action, i.e. what it does, and how it works" (Andreasen et al. 2015). An organ is thus a 'function carrier' and can be considered as the means for realizing the functions in the product. The organ domain is therefore an abstract representation of the product and does not capture the material and physical embodiment of the organs. That is reserved for the part domain.

The part domain articulates *how the product is built*, and thus models the physical components and their interfaces. Each element of the *part domain* is called a *part*, which interacts with other parts to realize the mode of action of an organ. It is the collective behavior of the parts that results in the intended behavior.

How does this research project relate to Domain Theory?

The Domain Theory provides a framework for reasoning about products from different viewpoints, which is useful when trying to understand the mode of action of a product, or when designing a product with a certain mode of action. It also provides a mental model of how the act of *designing* progresses iteratively between the three domains. This research project contributes with a qualification of interactions and interfaces, which may be applied in the Domain Theory between organs and the parts.

3.2.7 Theory of properties

Properties of a product arise as a result of interactions between system elements in a physical system. As stated earlier, behavioral properties of a system do not necessarily uniquely correlate with a specific structure of a product. Different structures and interaction patterns, may lead to the same behavioral properties.

What are types of properties of a system?

Hubka & Eder (1988) present a *Theory of properties* based on the idea that it is possible to define a complete list of properties, which can be designed into a technical system. See Figure 20.

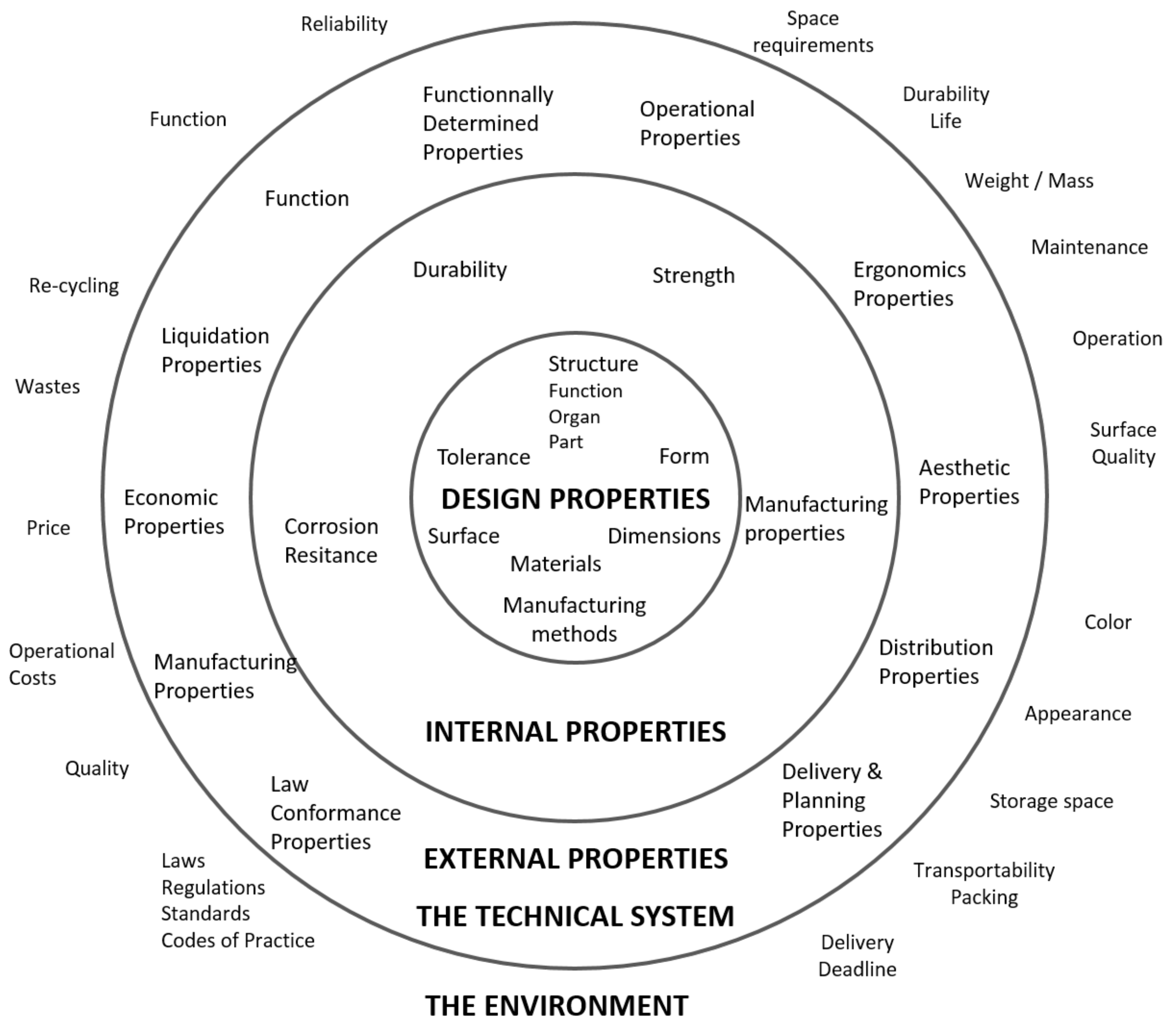


Figure 20 Classes of Properties and the relationships between them. Redrawn form (Hubka and Eder 1988)

The different properties are divided into design properties, internal properties, external properties (the technical system) and environmental demands.

This idea of a complete list of properties is later departed from by (Andreasen et al. 2015) because it is argued that properties are only partial viewpoints of a system and cannot be claimed to cover a complete set of properties. Instead Andreasen et al. (2015) classifies a products attributes into characteristics and properties.

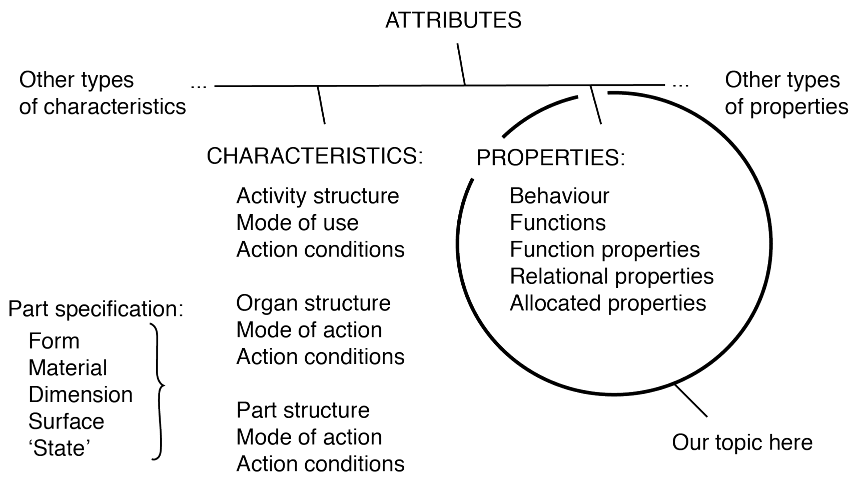


Figure 21 Classification of a product's attributes, specifically properties. Redrawn from (Andreasen et al. 2015)

Here, characteristics are defined as "*a class of structural attributes of products and activities determined by the synthesis of the design*" (Andreasen et al. 2015). This is in accordance with (Weber 2014) who defines a product's characteristics as the part structure, shape, dimensions, materials and surfaces, which can be directly influenced and manipulated by the designer.

Properties are defined as "*a behavioral class of devices' and activities' attributes, by which they show their appearance in the widest sense and create their relation to the surroundings*" (Andreasen et al. 2015). Weber (2014) concurs with this by defining properties as describing the product's behavior, e.g. function, weight, safety, reliability, aesthetic properties as well as manufacturability, assemblability, testability, environmental friendliness etc.

According to Andreasen et al. (2015) properties can further be classified into:

- *Behavior*: Actual behavior of product
- *Functions*: Intended behavior of product
- *Function properties*: Articulates the *goodness* of the function's realization
- *Relational properties*: Articulates the behavior of a product when used in its context
- *Allocated properties*: Articulates properties that customers, users, stakeholders, and society relate to products in a symbolic or devotional way, i.e. excitement, style, trend, hobbies, origin, brand etc.

The realization of these properties is a process of reasoning from issues, to requirements, to properties, and to characteristics of activity, organ, and part (Andreasen et al. 2015).

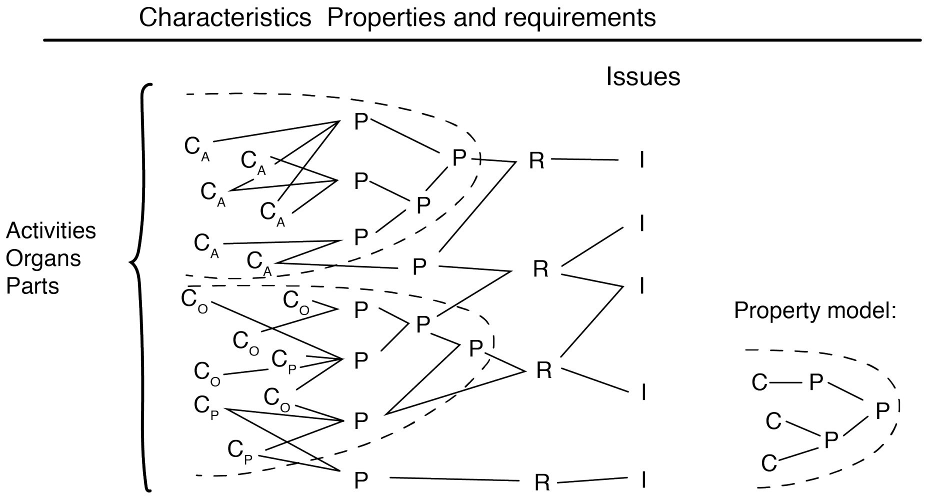


Figure 22 "The link model" illustrating how issues (I) are linked to requirements (R), to properties (P) and characteristics (C) of activities, organs, and parts. Redrawn from (Andreasen et al. 2015)

As Figure 22 outlines, requirements are textual statement articulating what the product should do. These requirements are hence closely linked with properties of the system. While the requirements should be solution neutral, the characteristics of the system describe the solution. A *specification* thus contains characteristics of the solution.

How does this research project relate to Theory of Properties?

Many companies have a fear of changing a component in an existing product because they do not understand the effect of such a change. In other words, they do not have property models showing why their product is realized as it is and hence what will happen if certain parts of it is changed. Only by systematically reasoning from external issues, setting requirements, defining the properties and conceptualizing a solution with certain characteristics is it possible to gain such an overview.

Defining interactions is all about capturing and 'freezing' the properties of a system and being consistent about realizing them as the system is decomposed and the complexity starts to grow. Having an understanding of what is meant by *properties* and *characteristics, requirements and specifications* as well as having a reasoning pattern is important for the description of how to apply this research during synthesis.

3.2.8 Trade-offs

Balancing trade-offs is an inherent part of any engineering design project. A trade-off may occur whenever two properties share the same characteristic. Thus changing the characteristic may improve one property while compromising another property (Andreasen et al. 2015). If it is not possible to find a suitable compromise between the two and thus balance the trade-off, one must come up with another concept, which alters the links (i.e. decouple) between properties and characteristics as depicted in Figure 22.

According Axiomatic Design (Suh 1990) one should strive for functional independence, i.e. no shared characteristics. Understanding the property models is thus a condition.

How does this research project relate to Trade-offs?

Trade-offs are part of any design activity. It is especially important in multi-technological projects that all multi-disciplinary trade-offs are exposed and decided upon due to the risk of lack of ownership. This is also the case when defining interactions and interfaces, because there may be parametric relations between them, i.e. changing the diameter of a hole (i.e. interface) affects several properties of an interaction such as the amount of material flowing, the rate at which thermal energy can be transferred etc.

3.2.9 Product architecture

Several definitions of product architecture exist. The following three have been selected for reference:

- *"(1) the arrangement of functional elements, (2) the mapping from functional elements to physical components, (3) the specification of the interfaces among interacting physical components"* (Ulrich 1995)
- "*The architecture of a product is the scheme by which the functional elements of the product are arranged into physical chunks and by which the chunks interact*" (Ulrich and Eppinger 2012)
- "*A product architecture is constituted by existing standard designs, existing design units, future standard designs and future design units. The architecture includes interfaces among the units and interfaces with the surroundings*" (Harlou 2006)

Common to all three definitions is the focus on interfaces between the various elements, i.e. components, chunks, units. Whereas the two first definitions articulate the mapping between the functional elements and the physical elements, the last definition is much more focused on a product variant perspective.

The product architecture is defined in the very early stages of product development where the product only exists as abstract functional drawings of models of how it is going to be realized into physical form. The choice of architecture has an impact on how the product will perform in its various life phases.

Ulrich (1995) proposes a typology of product architectures namely *integral* and *modular* architectures. A modular architecture has a *one-to-one* mapping between functions and physical "chunks" also known as modules. Modules are thus self-contained functioning elements. An integral architecture is in opposition a *non one-to-one* mapping. Both types have their advantages and disadvantages.

How does this research project relate to Product architectures?

As can be seen from the definitions, interfaces are a key part of any product architecture. The contributions from this thesis are thus intended to be used during the architectural phase of product development as stated in the research aim. Achieving a well-performing product architecture, requires a systematic approach to defining it, which is one of the contributions from this thesis. This research project applies to both integral and modular product architectures.

3.3 Theories related to product development

The following section will present theories, which are *concerned with the development* of the product (i.e. object) and not the description of the product itself. It is thus focused on the activity of doing design and the phenomena associated with this.

3.3.1 Theory of Dispositions

When developing a product, one must understand that the choices that are made in e.g. the design phase may have an effect on the products performance in other life-phases of the product. In other words, a designer must imagine how the product is going to be assembled in order to design a product, which is easy to assemble. This idea is named Theory of Disposition and was developed by (Olesen 1992). See Figure 23.

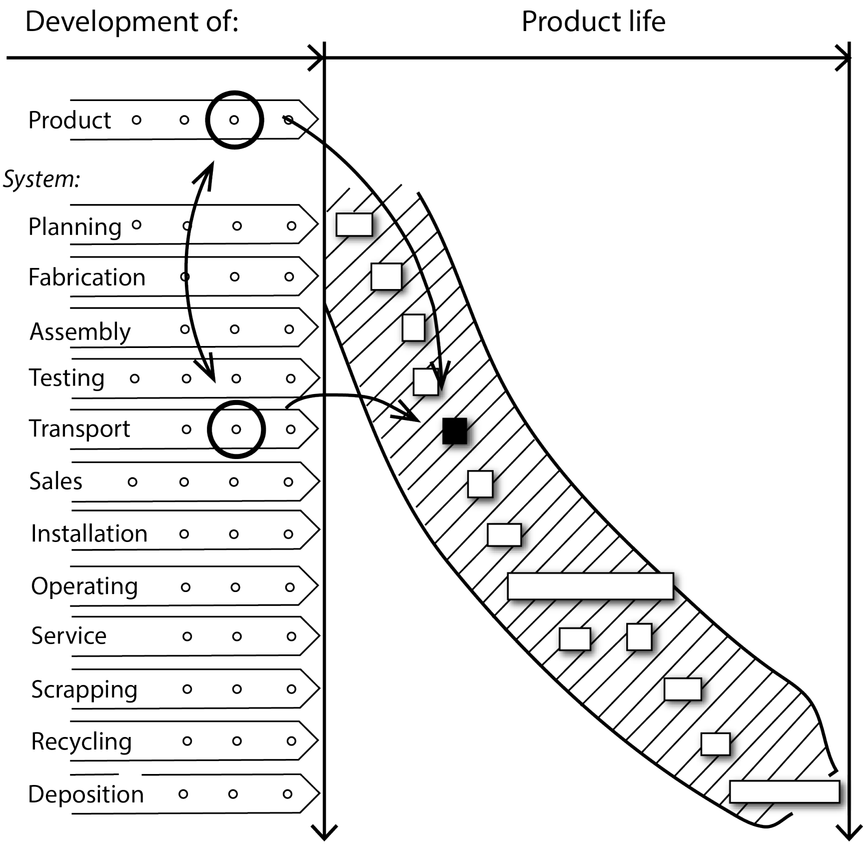


Figure 23 Theory of dispositions. Decisions made in the development of a product may have effects in other life-phases of the product

How does this research project relate to Theory of Dispositions?

When designing and embodying the interfaces of a system, the designer applies dispositional reasoning by reflecting upon how the interface can be manufactured, assembled, disassembled at the end of life etc. This reasoning might result in more interfaces being created or removed and thus *secondary interaction mechanisms* may arise as a result of a particular interface design.

3.3.2 Integrated Product Development

An influential book in Danish industry have been that of Andreasen and Hein (2000) where they focus on how to integrate product development across marketing, R&D and production. It articulates some of the fundamental characteristics of design such as “the advantage of dividing the project into phases, key point decisions, planning and collaboration between functional units in the company as well as exemplifying the importance of concurrent engineering” (Torry-smith 2013). The following illustration shows the concurrent activities of three typical functional silos in a company; Sales, Marketing, R&D, and Production. See Figure 24.

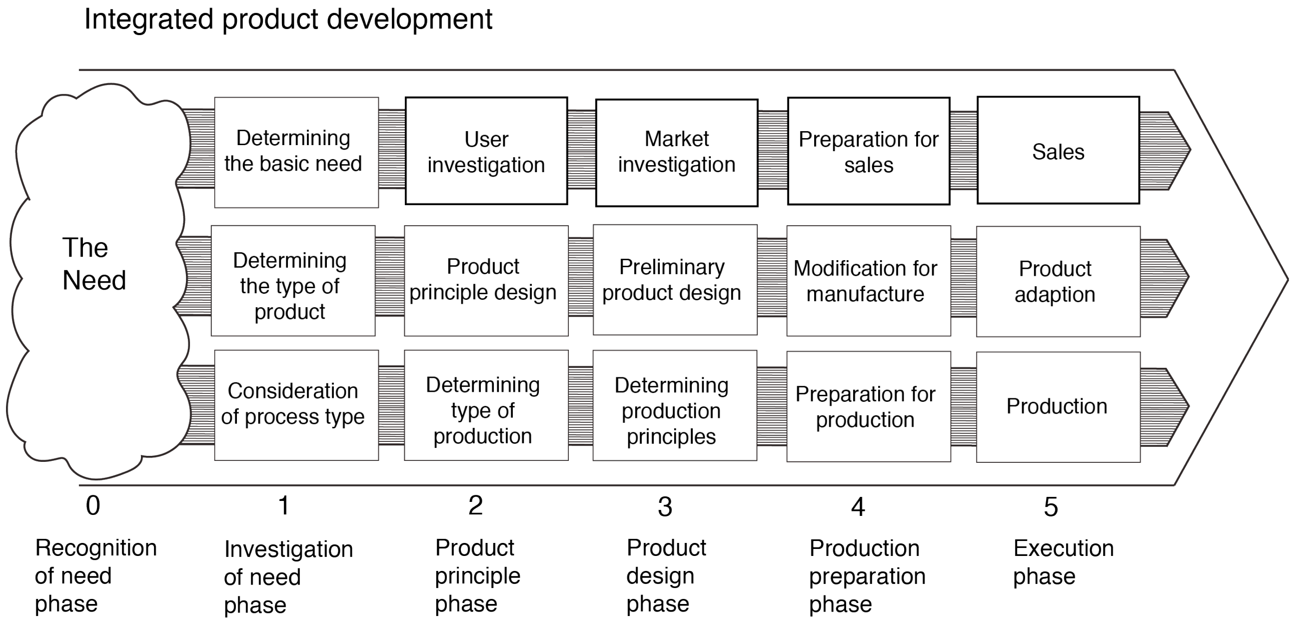


Figure 24 The Integrated Product Development model showing the concurrent sequence of tasks related to each function in a company as well as to the different development phases. Redrawn from (Andreasen and Hein 2000)

How does this research project relate to theory on Integrated Product Development?

This theory provides a basic understanding of how product development is ideally conducted in industry and the division of tasks between various functions in a company. The contributions from this research project will primarily be related to the early R&D task.

3.3.3 Systems Engineering

Systems engineering (SE) is a multi-disciplinary field of research and practice. SE is based on Systems thinking, which enforces one's awareness of the whole and how the parts within the whole interrelate (Haskins et al. 2006). SE is thus concerned with designing systems as a whole and not the constituent parts as such. An authoritative organization within the field of Systems Engineering is the *International Council on Systems Engineering* (INCOSE).

The following diagram illustrates the systematic top-down approach to developing systems. See Figure 25.

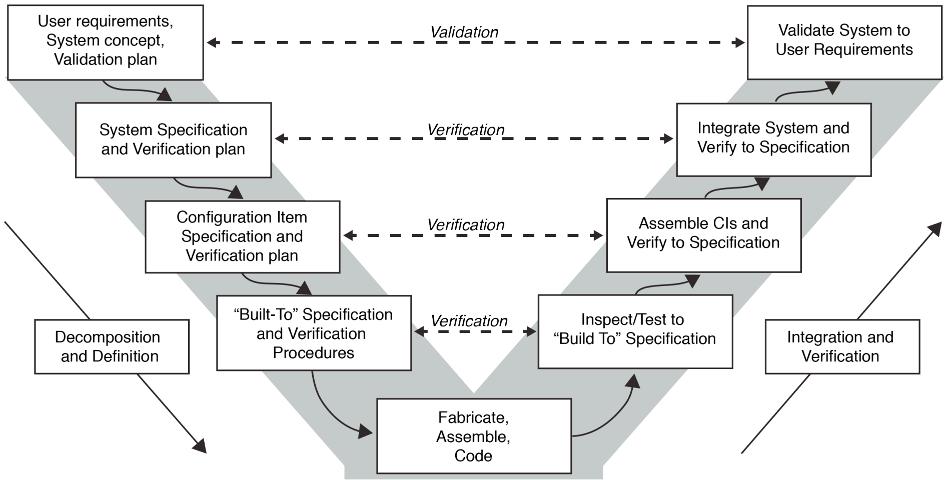


Figure 25 The Systems Engineering Vee model. Redrawn from (Dickerson and Mavris 2010)

The Vee model is read from left to right. The left side of the Vee is concerned with identifying and defining what the system should be able to do and translating this into a systems description and model. The system is first modeled from a functional perspective and then conceptualized into physical form and gradually decomposed into an appropriate level. Detailed design then designs the components and parts, which are then manufactured or coded. The right side of the Vee is concerned with testing that the manufactured items comply with the specifications (i.e. are we building the *product right*?) and that the behavior of the manufactures product comply with the intended use of the product (i.e. are we building the *right product*?) (Dickerson and Mavris 2010).

The Vee model is depicted in a sequential order, however in reality this process is much more iterative with a constant shift between synthesizing the solution, and analyzing the result.

How does this research project relate to Systems Engineering?

SE provides a systematic framework for designing complex systems. It outlines a top-down approach, which is useful to the application of this research. This research project thus contributes to the upper levels of the Vee, both in terms of decomposing the system, but also in terms of composing and testing the system as a whole.

3.3.4 Design structure matrix

The Design Structure Matrix (DSM) is a modeling method originally developed by (Steward 1981) capable of modeling any system by means of matrices, including technical systems. Just like a network-based diagram, as represented in Figure 15, a DSM consists of elements (i.e. components or parts), which are listed as both rows and columns, and relations between the elements (i.e. interactions or interfaces) represented by crosses in the matrix. In the paper by Pimmler and Eppinger (1994) they invent a rating system for interactions, which allows them to quantify the significance of an interaction on the product's functionality.

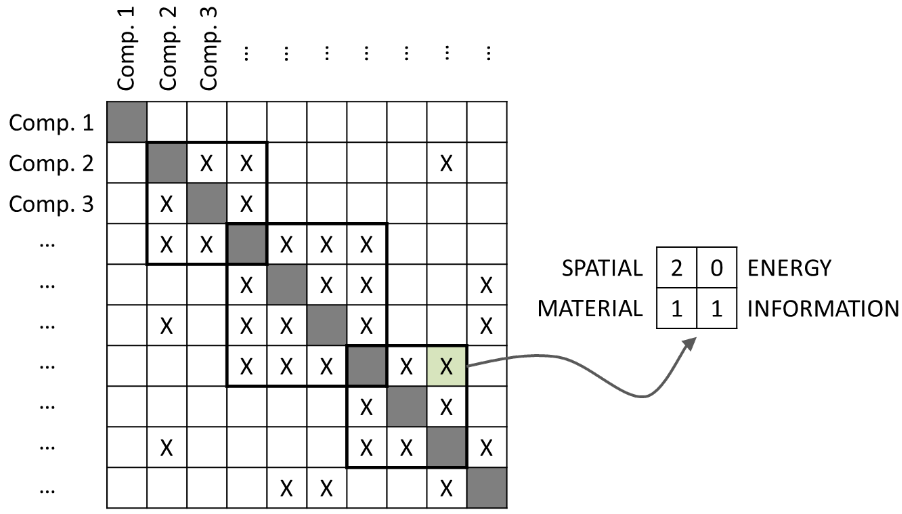


Figure 26 DSM of a generic product. Rows and columns both represent the components. The illustration is inspired from (Pimmler and Eppinger 1994)

With the extra information other researchers have attempted to apply computer algorithms to reorganize the matrix based on interrelations, e.g. automatic identification of modules. The matrix-based modeling method has over the years grown into a research community with dedicated conferences on DSM modeling (Eppinger and Browning 2012).

How does this research project relate to Design Structure Matrices?

Design Structure Matrix (DSM) as a modeling method is used actively in this research to model a system and to understand the complexity of the system. As such, it is a rather simple representation of a system, which can contain a lot of information. For novices or more experienced readers, a DSM may however seem a bit difficult to interpret at first. DSM is therefore a very powerful alignment method for the people involved in filling out the DSM, as well as for doing computational analysis of systems.

3.4 Physics

Physics belongs to the field of natural sciences and is concerned with predicting the natural behavior of the universe (Wikipedia 2016). Physics can generally be divided into *theoretical physics* which theoretically predicts natural behavior and *experimental physics,* which observes natural behavior by experiment, e.g. many of the theoretical predictions of Einstein has been experimentally proven today like recent discovery of the existence of gravitational waves (Abbott et al. 2016).

Physics is one of the oldest scientific disciplines dating back to the Greek philosophers and continues to amaze with new discoveries. Much of the physics we use today for describing everyday things was conceived from 17th century to early 20th century by what is today known as *Classical physics* and *Modern physics* (Wikipedia 2016).

In this thesis we apply a 20th century perspective on physics by basing our contribution on a central book called *Matter and Interactions* by (Chabay and Sherwood 2011). The book contributes to *contemporary* *physics*, where all physical phenomena are explained using a few fundamental principles. This *unification* into a few principles is key to deriving the *common language*, which is needed in multi-disciplinary engineering design.

Two fundamental principles, which are foundational for this research, in particular Paper C & D, are:

- *Law of conservation of momentum and energy*
  - i.e. any momentum or energy, which is gained by a system, is lost by its surroundings - in other words a zero-sum game
- *The fundamental Interactions*
  - i.e. all physical behavior can be explained using four fundamental interaction forces of nature; Gravitational, Electromagnetic, 'Strong' (aka. Nuclear force), and 'Weak' force

We will not go further into the theory behind these two fundamental principles, but merely refer to Paper C or Chabay and Sherwood (2011) for an in-depth treatment of these principles.

4 Results

The aim of Part 4 is to present the results of the research as documented in the four appended papers. Each description will feature the relevant research question, research method, research contribution including key figures and tables, and finally reflections on the contribution.

The research contributions presented in this Part 4 represents the result of a three year intensive project into the nature of *interactions* and *interfaces*. The original scope was to only look at *interfaces*, but not long after researching the literature and talking to industry experts was it clear that there is a discrepancy in the perception and understanding of the term *interface*, e.g. the meaning of an *interface* and an *interaction* are used interchangeably. Another observation was that there seem to be a causal relationship between an *interaction* and an *interface* due to the notion that it is not possible to design a suitable bridge (i.e. *interface*) if you don't know what is crossing the bridge (i.e. *interaction*) metaphorically speaking. It was therefore decided that in order to arrive at a rigorous and useful *interface* concept, we needed to derive an unambiguous and complete classification of *interactions*.

The purpose of Paper A in this thesis is to clarify the role of interfaces in new medical device development of product families through a case study. The purpose of Paper B is to motivate and clarify the phenomena concerning interactions and interfaces in engineering design through a literature review. Having collected this basis of understanding Paper C prescribes a framework containing a new classification of interactions derived from fundamental physics. In Paper D this framework is extended with an in depth definition of an interface as well as a prescriptive model of how to use the framework. Both Paper C and D contain an initial evaluation of the prescriptive support.

4.1 Paper A

Title: *"Enabling reuse of documentation in new medical device development: a systematic architecting approach"*

Conference: *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (IDETC\CIE DTM). American Society of Mechanical Engineers (ASME) (*Published 2015*)

Contributor: Second author (Equal work effort with first author)

Case study: 2 year empirical case study on development of arterial blood gas samplers (medical devices)

4.1.1 Associated research question

RQ1: What is the high-level role of interactions and interfaces in product family design in new medical device development?

4.1.2 Research method

In order to get insight into the role of interaction and interfaces in engineering design, we have conducted a case study in a medium-sized medical device company (Yin 2013). We investigated a specific challenge that the case company was facing: *How to reuse test documentation across product families of medical devices in order to minimize the effort related to releasing a new medical device?* Having this specific challenge as a framing of the investigation suggests that the result will only reveal *one* role out of presumably many different roles that interactions and interfaces may have in a real world project.

As theoretical background information, we present here a model inspired by Andreasen's work on property reasoning, the link model (Andreasen et al. 2015), which depicts our understanding of how properties are realized by components in a product. In this paper we elaborate by showing how these property models are reflected in the verification tests. See Figure 27.

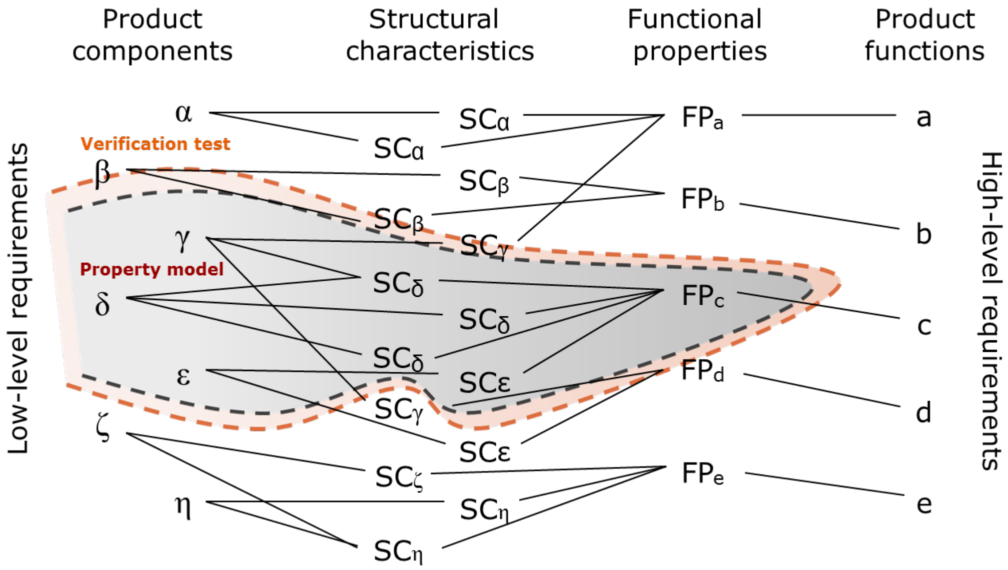


Figure 27 Breakdown-pattern of a system's attributes inspired by Andreasen's work on *Property Reasoning* (Andreasen et al. 2015). A product can thus be described by its functions, the properties of those functions respectively, the components and their characteristics that realize the functions. This illustration has been updated to improve communication and was presented at ASME IDETC/CIE DTM conference. See the original in the appended paper A

As can be seen from the figure, property models are discrete models connecting a particular functional property of a product to the components that realize it. When doing *verification tests,* what is actually tested and verified are these property models.

4.1.3 Research contribution

The investigation finds that in order to reuse test documentation across one or more families of product variants, one must modularize the products and define the interactions and interfaces between the variants and the invariant modules so well that if a given module configuration does not comply with the interaction and interface definitions, it can easily be discarded by inspection. If a new module configuration however lives up to the interaction and interface definitions, the verification tests may be reused with reference to the satisfied interaction and interface definitions.

The study also finds that if it is not possible for a new module configuration or a changed module to live up to the interaction and interface definitions, one may seek to decouple the property models spanning the invariant and variant modules so that the invariant modules (i.e. platform) have isolated properties, which may be tested for in isolation and thus reused across all relevant product variants. Again, as long as the interaction and interface definitions defining the boundary conditions to the invariant modules are fulfilled, functional and structural compatibility is ensured.

One challenge is to prioritize and understand the relevant property models of the product however another equally important challenge in the context of this research project is the question of *how do we describe interactions and interfaces well enough to confidently assure functional and structural compatibility?* Once the interaction and interface descriptions are used as a rationale towards the authorities, these documents are elevated to legal matters. The quality of the descriptions must therefore be extremely high. *But how can you describe interactions and interfaces when multiple engineering disciplines are involved? Is there a consensus across the various engineering domain about what the "correct" definition is?*

4.1.4 Reflections on contribution

- The use of interaction and interface definitions for ensuring reuse of test documentation is an example of a case where any misconception of the interaction and interface definition may have significant consequences on project cost, time-to-market or ultimately pose a risk to the health and safety of the end user. In this case, with such an influential role, there needs to be an unambiguous way of defining both interactions and interfaces
- Modularizing the product with the purpose of easing the documentation effort is just one out of many drivers of modularization. Therefore, the consequences of incomplete interaction and interface definitions may lead to ripple effects in many other domains as well, e.g. manufacturing, assembly, disassembly, use phase etc. where properties such as reliability, robustness, flexibility among other are important

4.2 Paper B

Title: *"Interface definitions in literature: A reality check"*
Journal: Concurrent Engineering – Research and Applications (*Published 2015*)
Contributor: First author

4.2.1 Associated research questions

- *RQ2: How are interfaces defined and perceived in literature?*
- *RQ3: What phenomena in multi-disciplinary product development are likely causes of problems occurring at interfaces?*

4.2.2 Research method

A systematic literature review has been performed initiated by a keyword search. This resulted in a vast amount of articles, which were first of all narrowed down by category (i.e. *engineering*) and language (i.e. *English*). To narrow down the results even further a review of the titles was performed and lastly a backward and forward search was done based on citations.

All definitions of interfaces were extracted from the papers, compared, and discussed up against four key issues in order to characterize the nature of an interface. The four key issues are:

1. Perception of the interface manifestation
2. Distinction between an interface (structural) and interaction (functional)

3. Perception of an interface as part of the elements in a system or as a design object
4. Types of elements used in the definition of an interface

Lastly a case example of a solenoid valve was used to reason about the phenomena associated with the activity of *interfacing*. The purpose of Paper B is not to identify or develop the most 'correct' or useful definition of an interface, but rather to clarify the discrepancies and discuss the implications to design practice (Parslov and Mortensen 2015).

4.2.3 Research contribution

The literature study revealed a significant difference in the perception of an interface. See Figure 28.

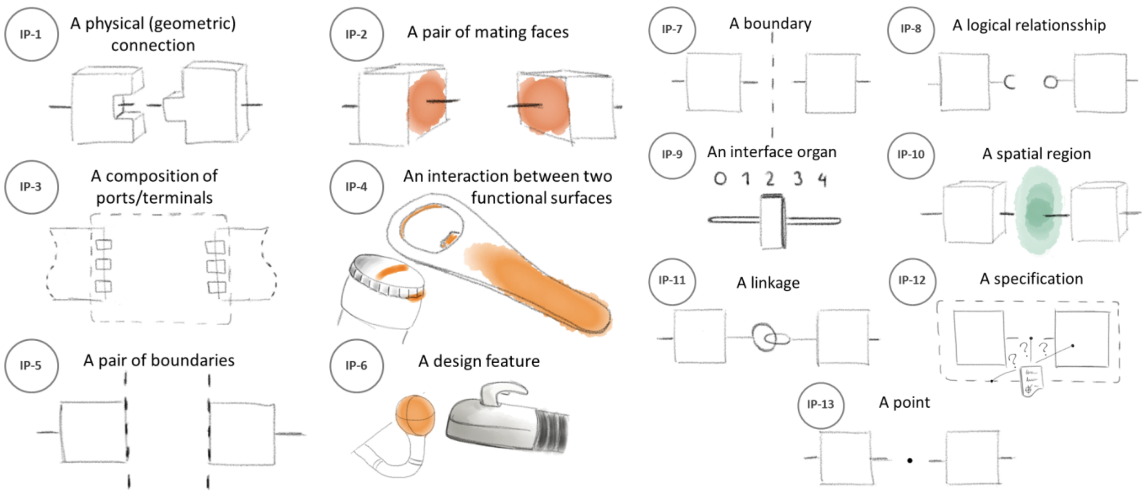


Figure 28 Illustrations of perceptions of interface manifestations as found in the literature (Parslov and Mortensen 2015)

The illustrations above are metaphorical in their representation of the various types of interfaces and not objective illustrations. As such, any human being may perceive an interface differently based on the provided definitions, which is exactly the point. Engineers from different disciplines work in different "object worlds" (Bucciarelli 1994) and thus may perceive common terms differently due to their difference in experience and conceptual world view. In other words, an interface as a term is ambiguous across different engineering disciplines, which may lead to miscommunication and ultimately rework (Parslov and Mortensen 2015).

A complete classification of the different perceptions of interfaces is presented in Table 5 tabulated against the four key issues.

Table 5 Overview of how different interface definitions relate certain perceptions of an interface to certain types of system elements (the perceptions consist of consolidated descriptions extracted directly from the definitions) (Parslov and Mortensen 2015)

		Perceptions of an interface												
		Part of element						Design object						
		Physical	Dual viewset (physical or functional)											
		A physical (geometric) connection	A pair of mating faces	A composition of ports/ terminals	An interaction between two functional surfaces	A pair of boundaries	A design feature	A boundary/a boundary area/a plane	A logical relationship	An interface organ	A spatial region	A linkage	A specification	A point
	Naming of elements	IP-1	IP-2	IP-3	IP-4	IP-5	IP-6	IP-7	IP-8	IP-9	IP-10	IP-11	IP-12	IP-13
Systems language	Systems/ subsystems/ elements/ entities/items		Sellgren and Andersson (1998); Blackenfelt and Sellgren (2000)	Grady (1994)	Sellgren and Andersson (1998)	Rahmani and Thomson (2012)	Lalli, Kastner and Hartt (1997)	Kapurch (2007); United States Department of Defense (2000, 2008); Liang and Paredis (2004); Grady (1994)	Rahmani and Thomson (2012)			Mikkola (2001)		Ullman (1992)
Function language	Functional surface/ functions/ functional units/area/ functional area				Sellgren and Andersson (1998)		Lalli, Kastner and Hartt (1997)	Kapurch (2007); ISO/IEC 2382-1:1993 (1993)		Buur (1990)				
Structural language	Modules	Lam and Shankar (1994)	Blackenfelt (2001)		Scalice, Andrade and Forcellini (2008)	Miller and Elgård (1998)						Mikkola JH (2001)	Baldwin and Clark (2000); Hoffman (1990)	
	Components/ parts/body	Ulrich (1995); Prasad (1997)	Sellgren and Andersson (1998)	Liang and Paredis (2004)	Scalice, Andrade and Forcellini (2008)						Van Wie, Greer, Campbell, et al. (2001)	Jarratt, Eckert and Clarkson (2004)		
Misc	Regions/groups							Ullman (1992)						Ullman (1992)
	Environment	Lam and Shankar AU (1994)									Van Wie, Greer, Campbell, et al. (2001)			
	[No mentioning of element]							United States Department of Defense (2001)					Prasad (1997)	

The main points from this contribution are:

- The literature review reveals a lack of consensus of the definition and meaning of an interface with respect to three of the four key issues:
  - 13 different perceptions of interfaces were discovered (key issue 1)
  - The majority of the literature state, that an interface can be viewed from both a functional and a structural point of view. According to most authors, functional interfaces occur at structural interfaces. (key issue 2). The term interface is such a common word in engineering and in disciplines outside of engineering that the diversity of people using the term is vast and thus the meaning of the term is subject to much interpretation
  - Around half of the authors consider an interface to be part of the elements indicating that it is composed of two entities. The other half considers an interface to be a design object in itself thus indicating a symmetric concept that separates the elements, rather than being part of the elements
  - The naming of the elements used in conjunction with the definitions of interfaces falls into three overall categories; Systems language, Functional language, Structural language. (Key issue 4). In total 15 different names for denoting the elements were used

Paper B ends with a case example of a solenoid valve used to regulate fluid flow in a blood gas instrument. The purpose of the case example is to discuss the phenomena related to the activity of *interfacing* in engineering design. The solenoid valve was thoroughly examined over the course of 6 months in order to fully grasp the physical phenomena. Some of the examining methods were:

- Electrical and mechanical test of remanence
- Electrical test of the influence of air gap vs. available pulling forces
- Surface roughness measurement
- Scanning electron microscopy and optical microscopy of surface irregularities
- Dimensional measurement and tolerance analysis
- Recording of signal characteristics with oscilloscope
- Energy-dispersive X-ray spectroscopy

The main points from the case example are:

- The example revolves around a solenoid valve where the solution principle involves an electromagnetic field that actuates a moving metallic anchor and thus realizes the overall function of opening and closing a flow of gas
- It is concluded that some interfaces may deserve greater attention than others because limited resources in projects doesn't allow for equal attention
- The discussion suggests three scenarios that qualifies an interface to receive greater attention:
  - *An inter-modular interface.* This interface is critical because it carries a lot of the overall functionality of the product stemming from the interaction between the two modules.
  - *An inter-disciplinary interface.* Such an interface may be subject to misinterpretation due to the difference in conceptual world views across the engineering disciplines
  - *A combination of an inter-modular and inter-disciplinary interface*

4.2.4 Reflections on the contribution

- An interface is a *conceptual construct* of engineering design that proves its usefulness once deployed in practice. It is defined to support engineers in describing a product. It is therefore not an exact scientific concept that can be observed and predicted. The purpose of the paper is to expose the differences of opinion through a systematic literature review of definitions and discuss the implications – not to favor one definition over another
- The four key issues relate to the *nature of the interface as a concept* and therefore serve the purpose of characterizing the different definitions. The key issues do therefore not speak about the phenomenon of *interfacing*, meaning that the treatment of definitions does not articulate *why* there is this difference in definitions relative to the engineering design context, but rather state the obvious factual difference. It is the understanding of the phenomena concerning the activity of interfacing that allows for an assessment of *why* one definition would be better than another. These *interfacing* phenomena are discussed in the case example but do not reflect back on the classification

4.3 Paper C

Title: *"Understanding Interactions in Complex Multi-Technological Products – A First Principle, Physics-based Theoretical Framework"*

Journal: Research in Engineering Design (*Submitted Feb. 2016, under 1st review*)

Contributor: First author

4.3.1 Associated research question

- *RQ4: How can interactions be classified using a physics-based first principles approach?*

4.3.2 Research method

The research method applied behind this paper is in itself a core contribution from this paper. The underlying research method is a *first principles approach* adopted from physics, meaning that the theory is deduced from the very fundamental theoretical 'building blocks' and built up from the bottom up. The first principle approach does not allow for any assumptions to be made when building up the classification, which allows for *mutually exclusive* classes (no overlap) of interactions and *collectively exhaustive* classes (no gaps, covers all technical disciplines) thus fulfilling one of the core objectives of this research project – to come up with a multi-disciplinary language, which can reduce ambiguity in the architectural decomposition of systems. Choosing this research method adds credibility to the internal validity of the *Interaction Framework*.

The theoretical framework is tested with 5 domain expert users. The evaluation study has been carefully designed to minimize bias. Future research must verify that the *Interaction framework* is useful in real-world projects.

4.3.3 Research contribution

The purpose of Paper C is to explain a new way of reasoning about interactions and interfaces in engineering design. The *Interaction framework* is exact from a physics perspective thus reducing ambiguity and foster interdisciplinary collaboration by providing a discipline independent language concerning interactions and interfaces.

One of the main contributions from Paper C is the distinction between INTERACTION and INTERACTION MECHANISM. It is the notion that INTERACTION, which is defined as physical properties consisting of momentum (translational & angular) and energy, are *facilitated* by an INTERACTION MECHANISM be it either FORCE or MATERIAL transfer. With this distinction we depart from the typical classification of interaction being *material, energy, information*, which is considered to be ambiguous because of non-exclusive classes.

While the concept of INTERACTION is a well-known, well-described concept of physics, the concept and classification of INTERACTION MECHANISM is novel in this cross-disciplinary field between physics and engineering design.

The following table shows the complete classification of INTERACTION and INTERACTION MECHANISM and how they relate to each other, see Table 6.

Table 6 Classification of INTERACTION MECHANISMS and INTERACTIONS and how they relate

						INTERACTION (EFFECT)	
	PRIMARY (ABSTRACTION)	SECONDARY (TYPE)	TERTIARY (LENGTH SCALE)	QUATERNARY (PATTERN OF MOVEMENT)	EXAMPLES USING FAMILIAR DOMAINS	**TRANSFER OF MOMENTUM** - (TRANSLATIONAL & ANGULAR)	**TRANSFER OF ENERGY**
INTERACTION MECHANISM (CAUSE)	FORCE	ELECTRO-MAGNETIC	MICROSCALE FIELD EFFECTS (PHYSICAL CONTACT)	RANDOM	*Thermal conductivity, stove, radiator etc.*	*NO*	THERMAL
				STATIC	*Assembly interfaces*	*NO*	*NO*
				CONSTANT MOVING	*Crane lifting container, compression of material, rotating shaft etc.*	YES	STRAIN
				WAVES	*Pistons, sound, earthquakes etc.*	YES	STRAIN, KINETIC
			MACROSCALE FIELD EFFECTS	STATIC	*Balloon on a jumper, permanent magnet/ electromagnet*	*NO*	*NO*
				CONSTANT MOVING	*Solenoid (constant current increase assumed)*	YES	ELECTRIC POT., MAGNETIC POT.
				WAVES	*EMR (i.e. sunlight, x-rays, UV-light, induction etc.)*	YES	KINETIC, ELECTRIC POT., MAGNETIC POT.
		GRAVITATIONAL	-	STATIC	*Earth's field (approx.)*	*NO*	*NO*
				CONSTANT MOVING	*Black holes with constant mass gain*	YES	GRAVITATIONAL POT.
	MATERIAL TRANSFER	-	ELEMENTARY PARTICLES (MICROSCALE)	CONSTANT MOVING	*Electricity, electrolysis, osmosis, diffusion etc.*	YES	KINETIC, GRAVITATIONAL POT., ELECTRIC POT., MAGNETIC POT.
			BULK MATERIAL (MACROSCALE)	CONSTANT MOVING	*Hydraulics, pneumatics, advection, etc.*	YES	CHEMICAL, THERMAL, STRAIN, KINETIC, GRAVITATIONAL POT., ELECTRIC POT., MAGNETIC POT.

Table 6 can be read in two ways depending on what type of reasoning is used – *synthesis* or *analysis*. When synthesizing a system one may reason about what type of INTERACTION is needed to obtain the intended function of the product and then use the table to see what type of INTERACTION MECHANISMS that may facilitate this particular transfer of INTERACTION. *Synthesis* thus prescribes a clock-wise read of Table 6.

When analyzing an existing current product by means of e.g. reverse engineering, one may reason backwards from first looking at a particular INTERACTION MECHANISM and then use the table to understand what INTERACTIONS are facilitated by this particular INTERACTION MECHANISM. *Analysis* thus prescribes a counter-clock-wise read of Table 6.

The main idea is that, often times, incompatibility problems occur, because certain INTERACTIONS or INTERACTION MECHANISMS are forgotten. With the Interaction Framework, the system architect or designer is presented with the complete overview, thus forcing them to actively exclude a certain INTERACTION or INTERACTION MECHANISM rather than relying on the individuals experience and carefulness.

The *Interaction Framework* was tested with 5 domain expert test participants (TPs) with various years of experience and educational background. While the number of data points is not enough to conclude based on statistical evidence, the initial evaluation does indicate a positive effect. Prior to being presented with the framework, the expert TPs with the most experience (TP4 and 5 collectively had 73 years of experience) identified more interactions than the less experienced experts (TP1-3 collectively had 29 years of experience). However, upon the introduction to the framework this difference seemed to be evened out, thus reducing the influence of experience. See Figure 29.

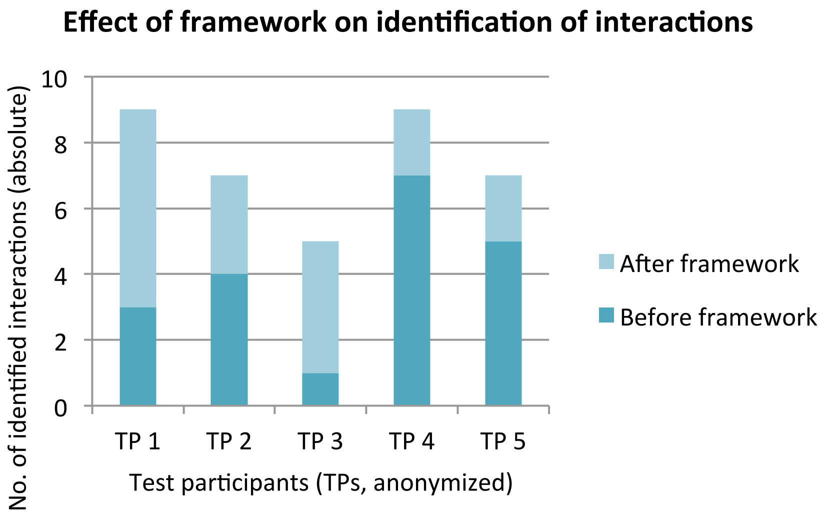


Figure 29 This chart displays the number of identified interactions per TP, both before and after they were exposed to the framework. NB. The "after framework" is a count of the added identified interactions

The TPs of the framework added on average 85% more interactions once having been exposed to the framework. This needs further verification through a real-world project. Another result was the fact that the TPs seemed to feel more comfortable thinking outside their own area of technical discipline, based on observations during the test. The TPs were asked to think aloud during their task execution in order for the researcher to tap into the reasoning pattern of the TP.

4.3.4 Reflections on the contribution

- The *Interaction Framework* is a significant contribution to the field of engineering design, because all previous attempts to come up with useful interaction classifications, in our opinion, have been based on *convenience* and therefore with overlapping and incomplete classes, which we believe introduces *ambiguity* to the architectural decomposition of a system
- The *Interaction Framework* is *a mindset* for reasoning and speaking about interactions and interfaces, which intends to bridge the language barrier between the different engineering disciplines. Because the framework builds on contemporary physics, which seeks to unify concepts to a few fundamental forces and mental models, we have been able to derive a common language and mindset concerning interactions and interfaces that is useful in multi-disciplinary product development
- Paper C does not contribute to characterizing a *design situation*. The sole purpose is to articulate *what* an interaction is and <u>not</u> *how* it emerges in a design situation or *who* that uses it
- The validity of the *Interaction Framework* is justified by the rigorous *first principle* approach to deriving the framework. The contribution is under peer-review with experts in both *engineering design* as well as *physics*
- The *usefulness* of the *Interaction Framework* to engineering design practice must be further verified by applying it in a real-world project. However, the initial evaluation indicates a positive effect in terms of capturing more interactions in general (i.e. applicability), including the ones outside the TPs area of technical expertise
- It is up to the user of the *Interaction Framework* to assess the relevance of each INTERACTION MECHANISM and facilitated INTERACTION case by case. The framework merely presents the options

4.4 Paper D

Title: *"Defining Interactions and Interfaces in Complex Multi-Technological Products – A Multi-disciplinary, Physics-based Approach"*

Journal: Research in Engineering Design (*Submitted feb. 2016, currently under review*)

Contributor: First author

4.4.1 Research question

Paper D addresses research question 5 and 6:

- *RQ5: How can an INTERFACE be defined and characterized, based on the understanding from the Interaction Framework?*
- *RQ6: How can the Interaction and Interface Framework be applied in practice to support complete and consistent INTERACTION and INTERFACE specifications?*

4.4.2 Research method

The contributions from this paper have been derived through logic reasoning from the physics-based framework. The 8-step architecting approach is developed based on the author's understanding of the phenomena inherent in engineering design, meaning that the classes which we arrive at are at an abstraction level applicable to engineering design. Examples of such phenomena are aspects of scale, abstractions, iterative development and gradual knowledge build up.

4.4.3 Research contribution

The aim of Paper D is twofold; first of all to summarize the *Interaction Framework* and extend it with a treatment of INTERACTION requirements/specifications and the concept of an INTERFACE. Second of all, the paper explains *how to use the framework* during the architecting of a system.

We introduce the notion of *Design input, Design Process, Design output* as a model for implementing the *Interaction and Interface Framework* in practice. The purpose is to stress the fact that the framework is used in the design process, which then outputs a document classified as design output. This output is then input to the next design process. See Figure 30.

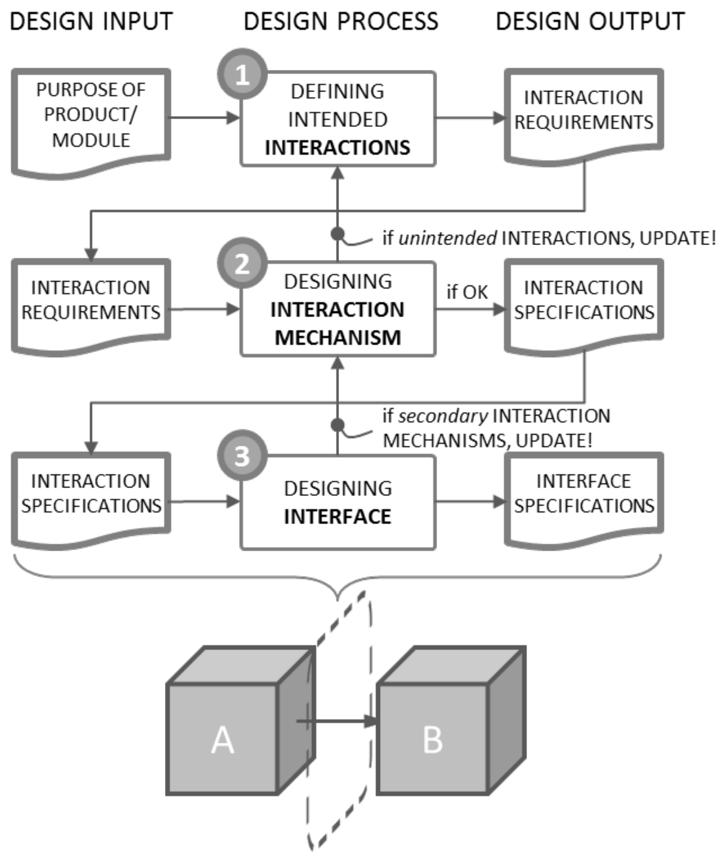


Figure 30 Illustration of the high-level INTERACTION and INTERFACE design process, starting at step 1. The arrows merely show the reading direction

Some companies are used to thinking of requirements as input to a process and specifications as output. However, the problem with this way of thinking is that *writing requirements* as a process may have a document containing requirements as an output in which case it would be called a requirement specification document, which may be misleading. By using the model of design input, design process, design output we may reduce ambiguity around the application of the process.

The high-level synthesis process therefore prescribes that the *intended* INTERACTIONS are defined first thus becoming input to the design of the *INTERACTION MECHANISMS* that facilitate the *intended* INTERACTIONS. In this regard the paper proposes an *Interaction specification template* and an *Interaction Specification Wheel (ISW)*, which support the translation from INTERACTION requirements to relevant characteristics of INTERACTION MECHANISMS. These are then documented in the *INTERACTION specification document*, which act as design input to the INTERFACE design process. A feedback loop is added to account for any *unintended* INTERACTIONS that may arise as a result of the chosen *primary* INTERACTION MECHANISM.

A system is defined by a system boundary, which separates the system from its surroundings. The place where an INTERACTION passes the system boundary is called an INTERFACE.

A classification of INTERFACES is therefore proposed; *simple* and *complex* INTERFACE. The classification adheres to the concepts presented in the *Interaction Framework,* while supporting the complex physical phenomena found in engineering systems such as friction, refraction etc.

The conception of a *simple* INTERFACE is that it is 'infinitely thin' and therefore does not transform an input to an output – it has zero function. This idea of 'infinitely thin' allows for the distinction between an interface and a component. Also, the concept applies to both functional and physical modeling viewpoints as commonly used in engineering design. See Figure 31.

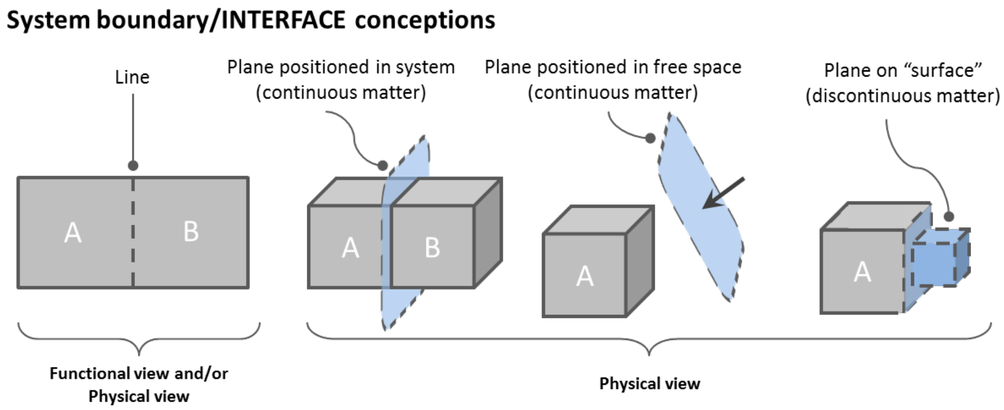


Figure 31 A system boundary may be perceived differently depending on the modeling viewpoint. Common to all conceptions are the notion of "infinitely thin" INTERFACE with zero function

In some special instances, we observe that *there is a transformation* of INTERACTION MECHANISMS acting at the INTERFACE, e.g. friction releases heat due to relative forced motion, refraction of light happens when light passes from one phase of matter to another. These special cases are classified as *complex* INTERFACES, which a basically an *abstraction* from the actual transformation and consider only the inputs and output from the *complex* INTERFACE. From an input/output perspective, the total sum must comply with the law of conservation of momentum and energy for both the simple and the complex INTERFACE. A feedback loop is added to capture any *secondary* INTERACTION MECHANISMS that may arise as a result of the chosen INTERFACE design.

An 8-step architecting approach is then presented, which prescribes how to use the Int*eraction and Interface Framework*. The 8-step approach builds on the same notion of design input, design process, design output. See Figure 32.

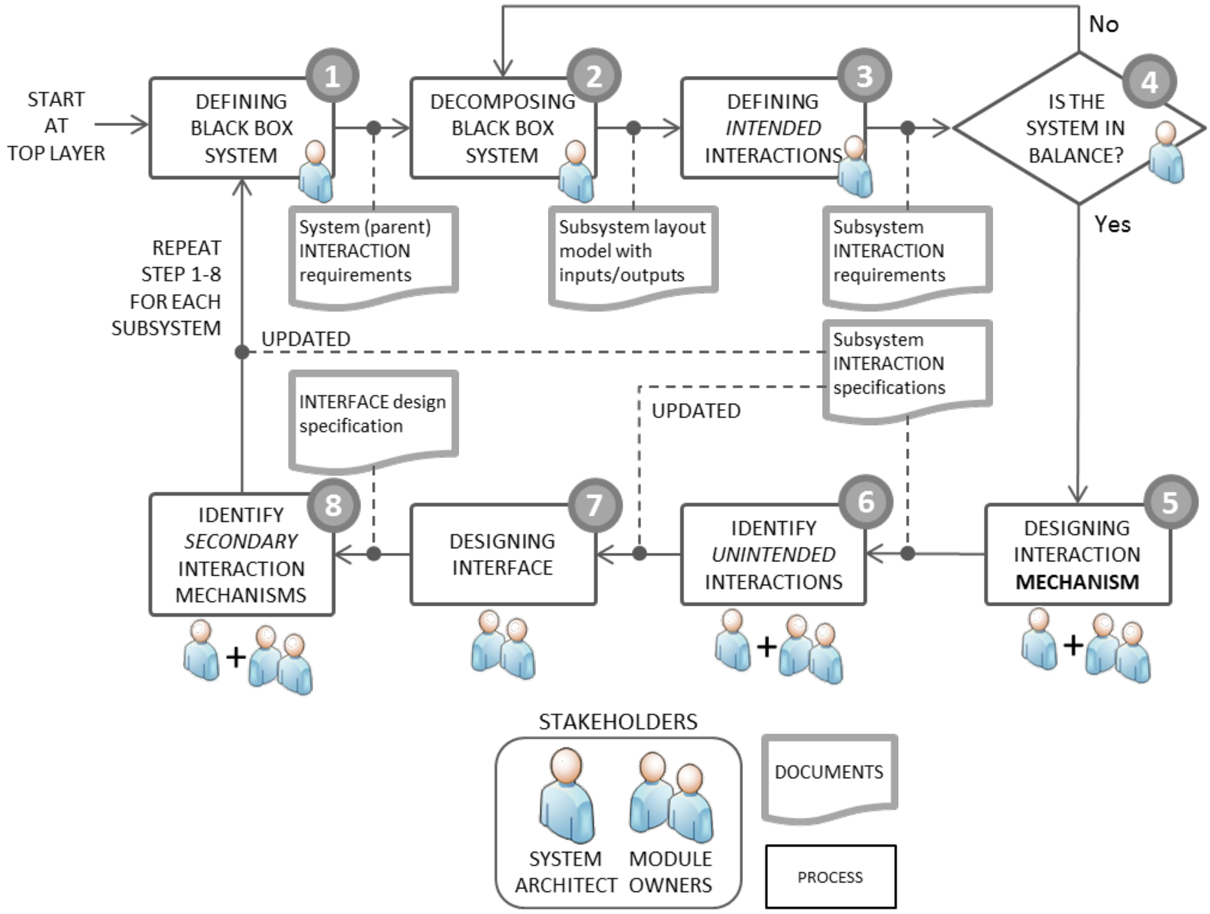


Figure 32 Illustration of 8-step architecting approach for applying the *Interaction Framework*

The 8-step approach is based on a top-down decomposition principle where the interactions and interfaces are described for each layer of decomposition starting with the top layer (i.e. the system of interest).

This particular approach enforces a systematic breakdown of the system, in which every aspect of the relations between the subsystems is actively considered. The approach also advocates for the constant shift in mindset between synthesis and analysis in which the system architect frequently must reflect on any added *unintended* INTERACTIONS or *secondary* INTERACTION MECHANISMS as a result of the maturation of the system. All 8-steps are repeated for each system or subsystem until a suitable level has been reached.

To support step 5, the paper presents the *INTERACTION Specification Wheel* (ISW), which is a physical tool at the size of the palm of a hand. The tool basically embodies the INTERACTION specification template and supports the system architect in classifying a certain INTERACTION by asking a series of questions to which only two outcomes exist. See Figure 33.

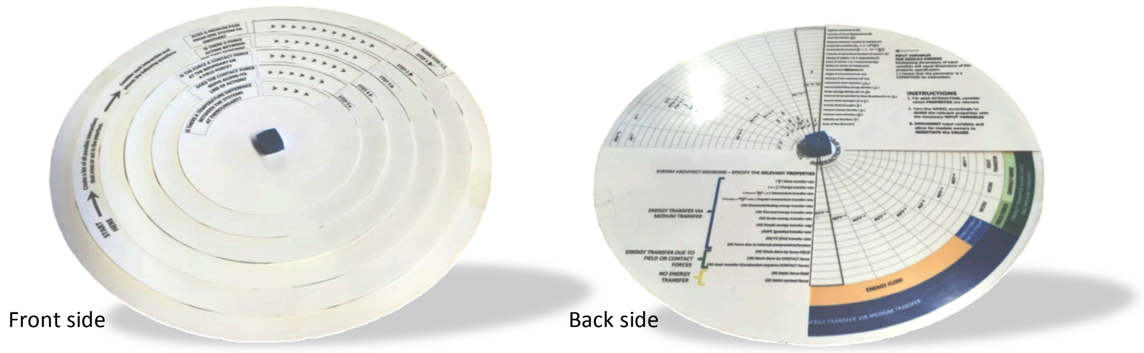


Figure 33 Prototype of the Interaction Specification Wheel (ISW) having two sides; the front side, which supports the system architect in classifying an INTERACTION MECHANISM and the backside, which supports the system architect in specifying the INTERACTION and INTERACTION MECHANISM

Finally, the Interaction Framework embodied by the ISW is tested in 5 domain expert user tests. The results show a significant improvement in the test participants' ability to consistently and unambiguously specify an INTERACTION MECHANISM. While there is not enough data to conclude on a statistical basis, this initial evaluation does indicate a positive verification and should therefore be followed up by a real world case study.

4.4.4 Reflection on the contributions

- With the *Interaction Framework* as a basis (see Paper C), it has been possible to extend the framework and derive a rigorous, multi-disciplinary concept of an interface, which does not describe the elements that are interfacing but rather the conditions necessary for an interaction to pass the interface. It thus succinctly separates an interface from a system element and considers it as a design object in its own right, which should be synthesized
- The 8-step architecting approach is the first formalized approach to systematically decompose and consistently define system boundaries from high-level interactions to interface embodiment in technical systems
- The validity of the extended concepts presented in this paper (i.e. requirements/specifications, INTERFACE) must be investigated in future research. It has not been possible within the time limit of this research project to investigate the *usefulness* of the concepts in a real-world project

5 Conclusion

The aim of Part 5 is to present the conclusions of this research project. It is structured into the overall research findings related to each research question (i.e. what have we learned?), the core contributions from this research (i.e. what were the tangible outcomes?), the evaluation of the result (i.e. why should we believe and trust in the results and how does it impact the recipients?) and suggestions for further research (i.e. what areas needs more research?)

5.1 Research findings

The following section will conclude on the research questions and describe what we have learned from this research project. The section is thus structured around the six research questions as introduced in section 1.4 Research questions. Research question 1-3 have a descriptive character and are about understanding the phenomena. Research question 4-6 have a prescriptive character and are about developing support.

5.1.1 Understanding the phenomena concerning interactions and interfaces

The purpose of the first part of the research project has been to first of all clarify the research objective by observing and analyzing industry practice. Secondly, the purpose has been to *investigate* and *understand* the phenomena concerning *interactions* and *interfaces* in new product development. The motivation for performing these two steps is based on the notion that *it is only possible to solve a problem, if you understand what the problem is*.

The first research question is thus an open, clarifying question:

RQ1
What is the high-level role of interactions and interfaces in product family design in new medical device development?

RQ1 have been addressed in Paper A through a 2 year empirical case study exploring the practice around new medical development.
Medical devices are challenged by heavy regulation and thus require extensive product documentation to market and maintain a product on market. The investigation therefore looked into how to reuse test documentation across product variants to leverage some of the cost associated with testing and documentation. With this fairly narrow scope of investigation we therefore expect to find a partial answer to RQ1.

From the case study it was found that verification tests are essentially verifications of whether the property models have sufficiently been realized in the actual product, and that property models express the properties of a product that result from interactions between its constituent components.
When seeking to reuse test documentation one must be able to rigorously justify that any difference there might be between the original tested product and the new variant does not affect the properties (i.e. the verification tests documents). It thus requires insight into the pattern of interactions that realize the properties to justify that no properties are changed.

Consequently, when changing or introducing a new component, the system architect may consult the interaction and interface requirement to understand whether the change will affect the common platform, or the properties spanning the platform and the variant components.

We can therefore conclude that interfaces and interaction requirements and specifications are instrumental in ensuring a sustained performance and therefore become the principle argument behind the *rationales* that are needed to argue in front of the authorities that a test can be reused across product families. In this case, the documentation of interactions and interfaces are elevated into a legally binding document of strategic importance because any inconsistencies may result in a warning letter from the authorities, thus potentially halving the value of a company (based on anecdotal information from a reliable source in industry). Defining interactions and interfaces unambiguously and consistently may therefore deserve the utmost attention in medical device development.

A natural question in continuation hereof may therefore be whether there is consensus concerning the definition and understanding of an interface and interaction, which leads us to the second research question.

RQ2

How are interfaces defined and perceived in literature?

This question has been answered by performing a systematic literature review on the definition of an interface within the engineering domain, see Paper B.

The literature review revealed that there is no consensus within the engineering design research community on what an interface is and how to conceptually perceive it.

- 13 different perceptions of the manifestation of an interface were discovered
- The majority of the literature states that an interface can be viewed from both a functional and a structural point of view. According to most authors, functional interfaces occur at structural interfaces. The term interface is such a common word in engineering and in disciplines outside of engineering that the diversity of people using the term is vast and thus the meaning of the term is subject to much interpretation
- Around half of the authors consider an interface to be part of the elements indicating that it is composed of two entities. The other half considers an interface to be a design object in itself thus indicating a symmetric concept that separates the elements, rather than being part of the elements
- The naming of the elements used in conjunction with the definitions of interfaces fall into three overall categories; Systems language, Functional language, Structural language. In total 15 different names for denoting the elements were used

The fact that there is no single universal definition of what an interface is problematic because it is commonly known that problems often occur at the interfaces during product development. Research question 3 therefore investigates the phenomena in multi-disciplinary product development related to interface problems.

RQ3

What phenomena in multi-disciplinary product development are likely causes of problems occurring at interfaces?

RQ3 is addressed in Paper B by reasoning based on a case example of a solenoid valve. The solenoid valve was thoroughly examined over the course of 6 months in order to fully grasp the physical phenomena. These analyses have clarified how the complex pattern of interactions results in a certain behavior of the system as a whole. It thus serves as a solid foundation for discussing how the different engineering disciplines are related through the physical realization of the solenoid valve.

Based on literature and this case example we can conclude that the various engineering disciplines have different mental models and conceptual understanding of products, which means that they speak different languages concerning what an interaction and interface is. As a consequence, decisions made in one engineering discipline, may propagate to other engineering disciplines as a result of the physical manifestation. In other words, due to a lack of common language and mindset, it is not possible for engineers within one discipline to reason about the physical effects that a decision may have in other disciplines.

Because it is not feasible to treat every interface in a system with the same level of detail, the discussion points toward three scenarios that qualifies an interface to receive greater attention:

- *An inter-modular interface.* This interface is critical because it carries a lot of the overall functionality of the product stemming from the interaction between the two modules
- *An inter-disciplinary interface.* Such an interface may be subject to misinterpretation due to the difference in conceptual world views across the engineering disciplines
- *A combination of an inter-modular and inter-disciplinary interface* where the risk of misconception is high due to the lack of common language and the severity of a potential failure is high because the inter-modular interface carries core functionality

From this research question it can be concluded that there are two phenomena that require attention:

- *Quality of the interaction and interface description* itself, which may be interpreted by various engineering disciplines
- *Quality of the communication concerning interaction and interfaces* in multi-disciplinary product development

This leads us to research question 4.

5.1.2 Developing a theoretical Interaction and Interface Framework

RQ4

How can interactions be classified using a physics-based first principles approach?

A key motivation for applying a first principle, physics-based approach has been the need for deriving a common language, which consists of mutually exclusive classes of interactions independent of engineering discipline yet covering all physical phenomena relevant to engineering design. Paper C therefore uses a significant amount of 'real-estate' to explain the fundamental principles of nature and to convey that common mindset, which is needed to bridge the language barrier between disciplines.

The following classification concludes a 1.5-year research effort into the nature and classification of interactions in the engineering design domain and represents the main answer to RQ4, see Table 7.

Table 7 Classification of INTERACTION and INTERACTION MECHANISM and a mapping of their relations

						INTERACTION (EFFECT)	
	PRIMARY (ABSTRACTION)	SECONDARY (TYPE)	TERTIARY (LENGTH SCALE)	QUATERNARY (PATTERN OF MOVEMENT)	EXAMPLES USING FAMILIAR DOMAINS	TRANSFER OF MOMENTUM - (TRANSLATIONAL & ANGULAR)	TRANSFER OF ENERGY
INTERACTION MECHANISM (CAUSE)	FORCE	ELECTRO-MAGNETIC	MICROSCALE FIELD EFFECTS (PHYSICAL CONTACT)	RANDOM	*Thermal conductivity, stove, radiator etc.*	*NO*	THERMAL
				STATIC	*Assembly interfaces*	*NO*	*NO*
				CONSTANT MOVING	*Crane lifting container, compression of material, rotating shaft etc.*	YES	STRAIN
				WAVES	*Pistons, sound, earthquakes etc.*	YES	STRAIN, KINETIC
			MACROSCALE FIELD EFFECTS	STATIC	*Balloon on a jumper, permanent magnet/ electromagnet*	*NO*	*NO*
				CONSTANT MOVING	*Solenoid (constant current increase assumed)*	YES	ELECTRIC POT., MAGNETIC POT.
				WAVES	*EMR (i.e. sunlight, x-rays, UV-light, induction etc.)*	YES	KINETIC, ELECTRIC POT., MAGNETIC POT.
		GRAVITATIONAL	-	STATIC	*Earth's field (approx.)*	*NO*	*NO*
				CONSTANT MOVING	*Black holes with constant mass gain*	YES	GRAVITATIONAL POT.
	MATERIAL TRANSFER	-	ELEMENTARY PARTICLES (MICROSCALE)	CONSTANT MOVING	*Electricity, electrolysis, osmosis, diffusion etc.*	YES	KINETIC, GRAVITATIONAL POT., ELECTRIC POT., MAGNETIC POT.
			BULK MATERIAL (MACROSCALE)	CONSTANT MOVING	*Hydraulics, pneumatics, advection, etc.*	YES	CHEMICAL, THERMAL, STRAIN, KINETIC, GRAVITATIONAL POT., ELECTRIC POT., MAGNETIC POT.

The classification is a significant step forward in the field of engineering design because of the following key characteristics:

- ***Mutually exclusive:*** The classes of INTERACTION MECHANISMS are deducted from first principles physics with zero assumptions. We can therefore conclude that the classes are not overlapping and thus *unambiguous*
- ***Collectively exhaustive:*** Apart from nuclear reactions, the classification of INTERACTION MECHANISMS and INTERACTION capture all physical phenomena relevant to engineering design and is thus *complete* from a physics perspective
- ***Multi-disciplinary:*** Because the starting point of the classification is physics, it has been possible to derive *a common language* that is applicable across *all engineering discipline* and capable of describing *all types of interactions* in *any product*

The use of this *Interaction Framework* may allow for better *communication* concerning interactions. It does however assume that it's users are capable of understanding the physics behind the framework as explained in Paper C.

The *Interaction Framework* also features an *Interaction Specification Template,* which supports the system architect in writing requirements and specifications for INTERACTIONS and INTERACTION MECHANISMS. See Paper D, appendix A. Using the *Interaction specification template,* the system architect may guarantee that the requirements and specifications are:

- ***Consistent***: Consistency is ensured using *dimensional analysis,* meaning that the requirements and specifications are neither under- nor over-constrained
- ***Complete***: The system architect may regularly consult the *complete* classification and actively decide what to include or exclude thus enforcing completeness in the requirements and specification

Using the *Interaction Framework* and the *Interaction specification template*, the system architect may therefore achieve an *unambiguous* description of interactions during the architectural decomposition of a system or product, which may lead to fewer integration errors.

Having defined the concept of an interaction, the following research question 5 investigates how to define the concept of an interface based on this understanding.

RQ5
How can an interface be defined and characterized, based on the understanding from the Interaction framework?

The concept of an interface is not an exact and measurable 'thing' of nature. Rather it is a concept that allows us to speak about the relations between elements of nature. Because of the *Interaction Framework* it has been possible to define an interface, which is applicable across any engineering discipline in its abstract form and captures the physical aspects of the mechanical domain in its concrete form. An INTERFACE can be defined as:

- The *conditions* necessary for an INTERACTION MECHANISM to occur (abstract form)
- The physical features that embody the INTERFACE and physically carry the INTERACTION MECHANISM (concrete form)

The INTERFACE *conditions* are summarized in the following Table 8.

Table 8 List of INTERFACE conditions derived from the choice of INTERACTION MECHANISM

INTERACTION MECHANISM	INTERFACE conditions
EM* MICROSCALE FIELD EFFECTS	Impermeability to matter, resistance
EM* MACROSCALE FIELD EFFECTS	Permeability to electromagnetic field force
GRAVITATIONAL FIELD EFFECTS	N/A**
ELEMENTARY PARTICLES (MICROSCALE MATERIAL transfer)	Permeability to elementary particles, conductivity
BULK MATERIAL (MACROSCALE MATERIAL transfer)	Permeability to bulk material, openness or absorbance

A key observation here is the fact that the conditions do not describe the *elements* that are *interfacing* but only characterizes the interface plane itself. These key characteristics of an INTERFACE support the gradual maturation from abstract to concrete.

Other characteristics are:

- Modeled and perceived as an 'infinitely thin' plane, see Figure 34
- Zero function meaning there is zero transformation of input to output from the interface

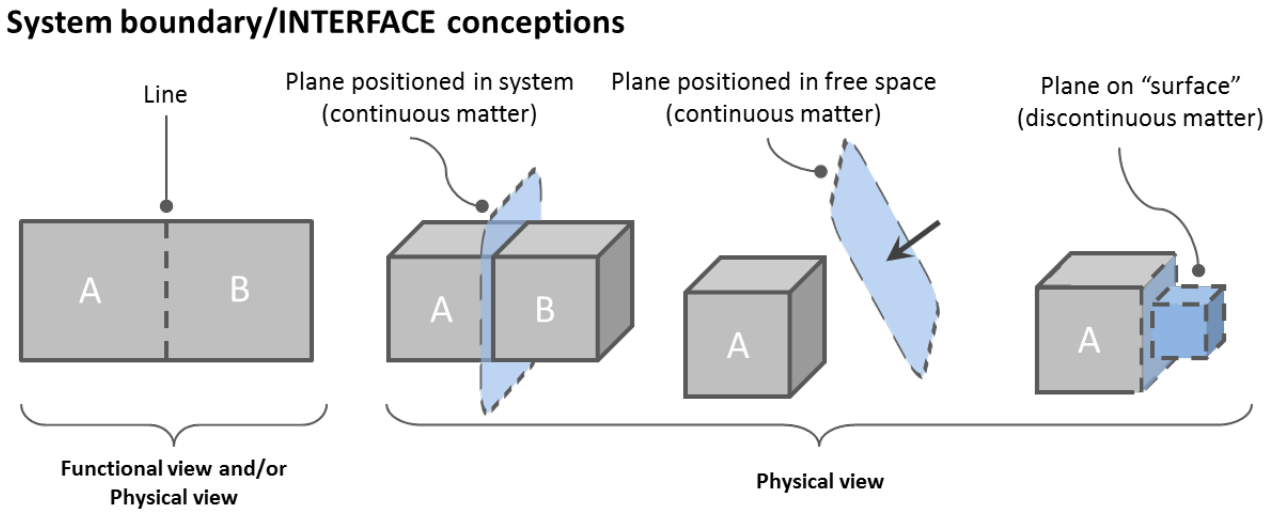


Figure 34 A model of how to perceive an INTERFACE in various modeling views, functional and physical

This way of modeling and perceiving the interface is compliant with both a functional and a physical modeling viewpoint.

In some ***special cases***, function, and therefore transformation from input to output, does occur across an interface when viewed from a product scale perspective, e.g. friction generates heat, refraction bends light, edges create turbulence in moving fluids. From a product scale perspective, these interface transformations are perceived as instantaneous, while at a molecular scale it might be a gradual transition.
In order to capture these special cases we distinguish between:

- a *Simple INTERFACE* (i.e. zero transformation, zero 'thickness') and
- a *Complex INTERFACE* (i.e. transformation, very 'thin')

Both the simple and the complex interface classes must comply with the law of conservation of momentum and energy. In other words, what goes in must come out of the interface. The only difference from a simple to a complex interface is that the interaction may have changed nature, e.g. strain energy input, strain + thermal output.

Having defined and classified an INTERACTION, INTERACTION MECHANISM, and INTERFACE a natural progression is to look into how to apply these concepts, hence the following research question.

RQ6

How can the Interaction and Interface framework be applied in practice to support complete and consistent INTERACTION and INTERFACE specifications?

In order to apply the theory and concepts in practice, an 8-step architectural decomposition approach has been defined, which builds on a top-down approach to product design.

The 8-step approach describes a 'loop' of 8 tasks for one level of decomposition. When having performed the 8 steps, one may decompose each subsystem to the next level and repeat the 8 steps. This is done until a satisfactory level have been reached. See Figure 35.

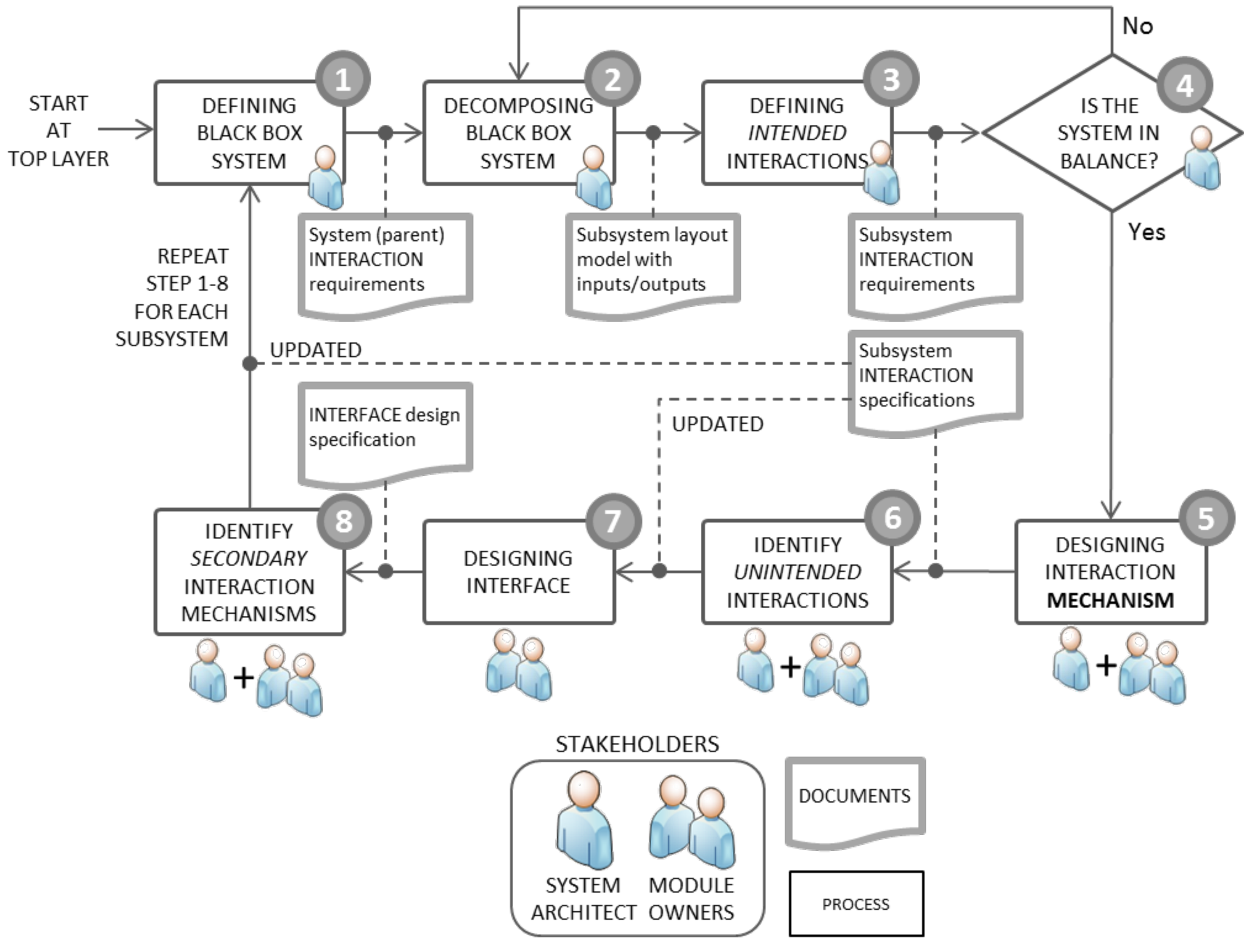


Figure 35 Illustration of the 8-step architecting approach used for applying the framework

We refer to Paper D for a detailed description of each step in the process. A key stakeholder is the *system architect* who is considered as the main user of the *Interaction and Interface Framework*. The *system architect* is an owner of the properties of the system and carries the overview of inputs and outputs between the subsystems. The *system architect* is in close dialog with the respective *module owners*, who are responsible for realizing the systems into a physical structure that fulfills the intended INTERACTIONS.

In a real-world design situation it is not always possible to foresee the emergence of an INTERACTION or an INTERACTION MECHANISM. Once the INTERACTION starts to be concretized by for example selecting the *primary* INTERACTION MECHANISM, *unintended* INTERACTIONS may emerge, which need to be accounted for in the requirements and the specifications. The same is true for INTERFACE design, where new, *secondary* INTERACTION MECHANISMS may emerge.

We therefore distinguish between:

- *Intended* INTERACTIONS: Those that serve the purpose of the system
- *Unintended* INTERACTIONS: Those that are inherently facilitated by the chosen INTERACTION MECHANISM but do not initially serve the purpose of the system
- *Primary* INTERACTION MECHANISMS: Those that are the primary mean of facilitating the intended and unintended INTERACTIONS
- *Secondary* INTERACTION MECHANISMS: Those that emerge from the embodiment of an INTERFACE and thus support the INTERFACE in transferring the primary INTERACTION MECHANISM

Capturing *unintended* INTERACTIONS and *secondary* INTERACTION MECHANISMS requires iteration and several feedback loops as drawn in Figure 35. It is crucial that *unintended* INTERACTIONS or *secondary* INTERACTION MECHANISMS are discovered early where the cost of change is low. A single overlooked INTERACTION MECHANISM may be the difference between failure and success.
The classification in Table 7 is thus applicable for both synthesizing a desired solution (i.e. step 3, 5, and 7) and for analyzing the designed solution (i.e. step 4, 6, and 8).

In a synthesis situation, one may reason from intended INTERACTION (i.e. momentum or energy transfer) to possible INTERACTION MECHANISMS (i.e. force or material transfer) for facilitating the INTERACTION. In analysis, one may identify an INTERACTION MECHANISM and reason backwards to understand, which INTERACTIONS are facilitated. Consequently, navigating the classification in Table 7 can be articulated using the following questions:

- Synthesis:
    - *What possible INTERACTION MECHANISMS (i.e. force and/or material transfer) may facilitate a given INTERACTION (i.e. transfer of momentum and/or energy) across an INTERFACE?*
- Analysis:
    - *What possible INTERACTIONS (i.e. transfer of momentum and/or energy) may be facilitated by a given INTERACTION MECHANISM (i.e. force and/or material transfer)?*

In order to support the design process of step 3 and 5, we have developed a hand-held tool called the Interaction Specification Wheel (ISW), which supports the system architect in creating unambiguous and consistent requirements and specifications for INTERACTION and INTERACTION MECHANISMS. It represents the information contained in the Interaction Specification template, which can be found in Appendix A of Paper D.
The ISW also supports the system architect and the module owners in performing trade-off studies because it discloses some of the relations between characteristics of INTERACTION MECHANISMS and the facilitated INTERACTIONS.

5.2 Core contributions

This research project has led to several core contributions to the field of engineering design research. The core contributions consist of classifications, definitions, mental models, common mindset, templates, and an architecting approach. The core contributions are summarized below:

Core contributions (CC) to *engineering design*, specifically *theory of technical systems:*

CC#1 *Systematic literature review* including a *classification of interface perceptions*
CC#2 *Definitions* and *classifications* of *INTERACTION*, *INTERACTION MECHANISM*, and *INTERFACE*
CC#3 *Common language, mental model and mindset* for reasoning about interactions and interfaces
CC#4 *Interaction Specification Template* and *Interaction Specification Wheel (tool)* for consistently defining *requirements* and *specifications* for INTERACTION and INTERACTION MECHANISM
CC#5 *8-step architecting approach* with explicit focus on defining and decomposing interactions and interfaces in multi-technological systems

The following section will systematically evaluate the validity of the contributions from each paper (A-D).

5.3 Evaluation of the research results

In the following section the evaluation of the descriptive and prescriptive research will be treated differently. For the *descriptive* papers we propose three criteria, which characterizes the validity of the results. For the *prescriptive* papers we will apply *the Validation Square* (Pedersen et al. 2000), which is applicable for evaluating design theory or methods.

The evaluation is thus split into the *descriptive part* (i.e. Paper A & B) and the *prescriptive part* (i.e. Paper C & D) for ease communication.

5.3.1 Understanding the phenomena (descriptive, Paper A & B)

Both Paper A and B are about understanding and describing the phenomena surrounding interactions and interfaces from a practical and a theoretical perspective respectively. These contributions will be discussed according to the following criteria:

- *Validity:* How was the result conceived?
- *Completeness:* How well do the results answer the research question?
- *Generality:* How can the results be used to generalize beyond the case example?

Validity:

- Paper A: The collected data have been carefully triangulated meaning that various methods were used to extract data from various sources, e.g. case study, reverse engineering, interviews, document review etc. The results are thus valid under the conditions of the research question
- Paper B: A comprehensive, systematic literature review was executed in Paper B in order to classify interface perceptions. A case example of a solenoid valve was reverse engineered and analyzed using well-established theories on technical systems and product development in order to reason out the phenomena of perception and communication about interfaces in multi-disciplinary development. This case example has been discussed with multiple senior researchers and practitioners within the field of engineering design

Completeness:

- Paper A: The answer to the associated research question is complete because it provides an important role of interfaces in new medical device development. There may be other high-level roles that interactions and interfaces play in product family development of medical devices. While the presented results do answer the question and thus motivate the further research in this project, there may be a need for further research if the purpose is to disclose the total number of roles. However, for the purpose of motivating why this research project is important, the result seems sufficient
- Paper B: The classification of interface perceptions from Paper B may be considered as *complete* due to the underlying comprehensive and systematic literature study. Paper B further suggests a list of three types of interfaces, which are likely to need more attention, however these three types are purely theoretically based and based on the author's experience from industry. There may therefore be other reasons for prioritizing certain interfaces, which could have been disclosed from other case examples or other research methods. This list is therefore not exhaustive and complete, however does point out some important aspects in relation to the perceptions and communication of interfaces in multi-disciplinary product development

Generality:

- Paper A: The role of interfaces as legal elements in medical device development is most likely generalizable to other heavy regulated industries. In fact, whenever a manufacturing company sources some of the components from (sub-) suppliers the interface specification becomes a document of legal matter like a contract. The generality of this result thus supports the need for more research on the understanding of interfaces
- Paper B: While the interface perceptions as presented in Paper B are purely based on literature, it may be representative for how engineers in industry *think* of an interface. The literature was deliberately chosen to represent different engineering disciplines in order to get as wide a collection of definitions as possible. The question is whether these perceptions are representative of the industry practitioners' perceptions as well. This will require more empirical evidence to support. While the case example of a solenoid valve is rather specific it does capture some of the core phenomena, which characterizes larger systems as well, such as multi-technological, modular and integral. The results are therefore generalizable to other complex systems

5.3.2 Developing the support (prescriptive, Paper C &D)

Evaluation of the core contributions (CC#2-5) from both Paper C & D will be structured around the six statements from *the Validation Square* (Pedersen et al. 2000). See section 2.4.3 The Validation Square for a description of the evaluation method.

Effectiveness (*structural validity/internal consistency)*:

1) *Individual constructs:* The classification of INTERACTION and INTERACTION MECHANISMS are systematically derived from *first principles of physics*. This *logical deduction* ensures *mutually exclusive* classes thus reducing *ambiguity* significantly. The classification of an INTERFACE is logically derived from the interaction classification. The 8-step approach is both inspired by well-established architectural decomposition theory (Harlou 2006; Hölttä-Otto et al. 2014; Crawley et al. 2015) and

based on the authors' work experience. The 8-step approach is designed explicitly for defining interactions and interfaces unlike typical methods that focus on defining the elements. All of the above constructs underlying the contributions are accepted as valid

2) *Internal consistency:* The *logical deduction* of the *Interaction and Interface Framework* from *first principles* is outlined in a flow diagram in 2.3.2 Research methods. The deduction of the classifications does not rest on any assumptions. The *internal consistency* is therefore considered as rigorous and accepted. The *internal consistency* of the *Interaction specification template* is ensured using dimensional analysis. The structure and order of the 8-step approach is based on well-founded literature and the authors' experience with architectural decomposition. The 8-step approach is not as sequential as depicted, but rather should be executed in an iterative manner. The internal consistency of the 8-step architecting approach is therefore accepted
3) *Appropriateness of example:* The Interaction Framework, as presented in Paper C, applies to any technical system. A multitude of different examples are used during the paper to support the notion that the interaction Framework describes a fundamental phenomenon of engineering systems. A hair dryer is used as a case example to test the performance of the framework. The physical phenomena in the hair dryer are considered as similar to any other example problem relevant for the framework. The example problem of identifying and characterizing interactions at a given interface in a multi-technological system is appropriate because they represent a proxy for assessing the *completeness*, *unambiguousness*, and *level of common understanding* across engineering disciplines concerning interactions and interfaces. This has been assessed both quantitatively and qualitatively. A criterion for selecting a hair dryer as case example was that the test participants should understand the mode of action of the hair dryer to remove any baseline effects. 5 out of 5 test participants answered that they 'totally agreed' with the statement that they understand the mode of action. The limited number of data-points (5 test participants) only allow for an indicative result, which is not statistically significant. Further research must verify the effects of applying the framework in practice, both quantitatively and qualitatively. A potential risk, which cannot be rejected based on the limited amount of test data, is if the users of the framework do not understand the physics behind it as explained in Paper C and thus find the framework difficult or ambiguous to use. So despite being unambiguous from a physics-standpoint, the end users may find it more confusing. Which is then better; the existing classifications or the proposed one? This is a risk that can only be mitigated through further testing

Efficiency (performance validity/external relevance):

4) *Usefulness of outcome from example:* As stated in Pedersen et al. (2000) the "*metrics for usefulness are linked to the degree an articulated purpose has been achieved*", which differs depending on the viewpoint; industrial or academic. From the case example and based on the author's work experiences it can be concluded that the outcome (CC#2-5) is useful from an industrial perspective because it supports the *quality* of the *description* and *communication* concerning interaction and interfaces in multi-disciplinary products. From an academic perspective, this *Interaction and Interface Framework* is a useful foundation for new research on *model based systems engineering, multi-disciplinary product development, concurrent engineering*, functional modeling etc.
5) *Usefulness linked to applied theory, method or tool:* A real-life case study has *not* been performed due to time limitations. The framework has instead been applied in a controlled test, where the test participants were asked to complete certain tasks both before and after being introduced to

the framework, thus exposing the positive effect. The consistency of the framework has also been compared to the existing classifications of interaction using a case example.
This comparison suggests that the usefulness is indeed correlated with the framework

6) *Usefulness beyond the example:* Based on the five points explained above it is the author's believe, that the *core contributions (CC#2-5)* are indeed useful beyond the case examples. It will however require several case studies (Yin 2013) to prove the *external validity* of the theory in a wide range of case examples

5.4 Evaluation of the research impact

In the following section we discuss the implication of this research from an academic, industrial, and societal perspective. We will also address whether the results from this thesis satisfy the aim and objectives set up for this research.

5.4.1 Academic impact (Paper A – D)

The overall aim of this PhD project was to *support early-stage architecture based product development* through an explicit focus on *how to conceptually understand and model interactions and interfaces*.

Based on the results and the core contributions from this research project it can be concluded that the PhD project does satisfy the aim of the project by proposing a multi-disciplinary, physics-based *Interaction and Interface Framework*. There is however a need for further testing in real-life projects in order to fully prove the *usefulness* and thus *external validity* of this framework as stated in 5.3 Evaluation of the research results.

The theoretical objective of this thesis was to *expand the body of knowledge of engineering design research and systems engineering by providing knowledge about the existing paradigm concerning interactions and interfaces as well as extending the knowledge about reasoning in multi-disciplinary engineering design research.*

It can be concluded that the core contributions from this research project do *add to the body of knowledge of Engineering Design*, in particular the Theory of Technical Systems (Hubka and Eder 1988) by defining the nature of INTERACTION, INTERACTION MECHANISMS and INTERFACES in technical systems. The proposed theory, the *Interaction and Interface Framework*, distinguish itself from the existing paradigms, by being based on first principles of physics. This allows for a high confidence in the internal construct and consistency of the theory. The *Interaction and Interface Framework* is multi-disciplinary at its core and therefore enforces multi-disciplinary reasoning for use in engineering design.

It is the author's intention that the core contributions from this research project will clarify the apparent *fuzziness* of the terms *interaction* and *interface* in multi-disciplinary product development and that the framework may become the *de facto standard* for *understanding* interactions and interfaces in engineering design. Due to the neutrality of the framework towards the various engineering design research 'schools', and the close attention to definition of terms it may be possible to incorporate these contributions into the theoretical frameworks of various 'schools', especially because all 'schools' today apply the "*material, energy, information"*-classification.

The future of engineering design is most likely one, in which products are designed end-to-end in a software simulation environment, which may significantly reduce the cost of product development.

However, in order for the very early phases of product development, the *architectural decomposition phase*, to be part of this digital environment there is a need for an unambiguous and complete common language across any engineering discipline concerning interactions and interfaces, which supports the modeling of products from highly abstract, undetailed models to very concrete, detailed models. I am confident this is what we have achieved as a result of this research project.

5.4.2 Industrial impact

The potential application areas of the proposed contributions are vast. Any physical product gains it's properties from internal interactions and interactions with its environment and has interfaces in order to manufacture the product. However, interfaces are also the cause of much distress because most problems occur at interfaces in a system (Grady 1994; Kapurch 2007; Wheatcraft 2010; Buede 2012). Applying this *Interaction and Interface Framework* is likely to reduce the number of interface-related problems.

From a single company perspective, having a *multi-disciplinary mindset* and a *common language* to speak about interactions and interfaces across engineering disciplines will lead to higher quality of communication and descriptions of interactions and interfaces during development. This will most likely reduce the amount of rework associated with problematic interfaces and thus reduce development cost and shorten the time-to-market.

From an industry point of view interaction and interface documentation are central in ensuring proper collaboration between manufacturing companies and suppliers of subsystems, e.g. an aircraft consists of subsystems, which are developed and produced across multiple countries by multiple companies, and multiple engineering disciplines and finally assembled into a functioning whole. It is therefore essential that the interaction and interface descriptions are complete and unambiguous. Dividing and allocating ownership and responsibility may also gain from this explicit and systematic top-down approach to defining interactions and interfaces.

5.4.3 Societal impact

With this research project we expand the body of knowledge of multi-disciplinary engineering design, specifically the theory of technical systems, in order to achieve more productive designing. The outcome from this research project may therefore be taught at universities and in industry as the principle way of perceiving, understanding, speaking, communicating, and documenting interactions and interfaces in engineering design. This may lead to greater productivity and prosperity in society as a whole.

5.5 Limitations of the research and the results

5.5.1 Theory based versus problem-based research

The prescriptive contributions from this research project are based on logic deduction from theory, meaning that the role of the researcher is 'left out of the equation'. The challenge with a theory-based approach is however to prove that the theory is useful in reality or practice. It has not been possible within the time frame of this research project to perform one or more case-studies validating the usefulness of the contributions. This is subject to further research.

5.5.2 Software – a core part of complex systems

It is evident from present trends that the future of products will rely heavily on software for realizing their functionality. It is however also evident that software relies on platforms of hardware in order to be executed, e.g. sensors, computing power, actuators and auxiliary systems. This research project focuses primarily on the definition and perception of physical interactions and interfaces in hardware products and systems, which carry the information/signals executed by software. The research project does therefore not look at interactions and interfaces between modules or elements of code.

5.6 Suggestions for further research

5.6.1 Identifying 'critical' interactions and interfaces in cost-conscious projects

Due to limited resources in most product development projects it may not be possible to document all interactions and interfaces with the same level of detail. This research project suggests some general characteristics of interfaces that may deserve more attention (see Paper B and D) than others. But *how does one assess 'criticality' of interfaces in the early stages of product development when the uncertainty is high and the product very immature? What characteristics go into assessing 'criticality'?*

5.6.2 Organizational considerations

In this research project, the existence of a *system architect function* in the organization has been proposed. The system architect is intended to own properties of the system and systematically define and design interactions and interfaces between subsystems; however, there may be other organizational structures that prove more useful. There is strong evidence to support the fact that the structure of product architectures are reflected by the structure of the organization developing the product (Colfer and Baldwin 2010; Cabigiosu and Camuffo 2012; MacCormack et al. 2012). This phenomenon is called the 'mirroring hypothesis'. It is therefore reasonable to believe that the structure of an organization has influence on the adoption and execution of this *Interaction and Interface Framework*, which must be further investigated. *What is the suitable organizational setup for facilitating the adoption of the framework?*

5.6.3 Shift in allocation of resources

The formalization of interaction and interface as design object requires more development resources to be allocated in the early stages of product development – the architectural and conceptual stages. The underlying assumption is however that the total cost of development will be reduced due to fewer changes in the late stages of product development where the cost of change is high, see Figure 2. This assumption must however be challenged and investigated in several case studies in order to prove the feasibility of the framework. *Can one afford <u>not to</u> systematically define interactions and interfaces?*

5.6.4 'Vehicle' for operationalizing the Interaction and Interface Framework

Due to the substantial amount of information/data, which is produced during the development and documentation of complex products, there is a need for a software tool to manage this information. In most regulated markets like the medical device industry it is conditional that the documentation of the product development process is consistent and traceable in order to get market clearance and stay on market. With the introduction of this nuanced way of reasoning about interaction (i.e. INTERACTIONS, INTERACTION MECHANISMS) the amount of information in a development project will grow. Further research into system modeling methods and tools as well as product data and requirements management systems is needed in order to identify a suitable 'vehicle' for the theory into practice.

5.6.5 Other influencing factors on the adoption of the research

Other factors, which may influence the adoption of this research such as size of project (people and cost), severity of impact in case of product failure, technology maturity, uncertainty, organizational capability and maturity etc. may be investigated further. The chances of successful adoption and execution of the theory is highly dependent on a proper 'fit' with the intended context.

5.7 Concluding remarks

This PhD project has been a tremendous personal endeavor from which I have learned a lot about myself and the activity of doing research.

The following points are my reflections about research and what it is all about. Maybe this can be of inspiration to future PhD students:

- The first week at your desk is terrifying; the second week you get used to hearing your own voice; the third week you realize that you are not the only one in this situation and you fearlessly take on the challenge!
- Research is about learning to reason at a meta-level about knowledge creation
- Research is about articulating and systemizing intangible thoughts and ideas into a concrete, coherent whole, which can be understood and applied by others to create value
- Clarity is best achieved through dialog with peers. Discussions with yourself can only bring you so far
- Knowledge-acquisition is a continuous process. Make sure to set a 'definition of done' so that you are not working up against a moving target
- Be aware of balancing abstraction and detail in the project before spending too much time on details, which do not add value to the overall project
- For projects in Engineering Design; ensure that the timing between the company's activities and your project schedule is aligned. It may take close to 1.5 year or more before you can test out your main contributions in practice. But is the company setting ready for implementing it at that time? Timing requires planning
- You, as PhD student, have the final say. Seek inspiration and advice from supervisors, colleagues and peers, but trust in your own decisiveness

6 References

Abbott B, Abbott R, Abbott T (2016) Observation of gravitational waves from a binary black hole merger.

Andreasen M, Hein L (2000) Integrated product development. IFS Publications Ltd

Andreasen MM (1980) Machine design methods based on a systemic approach. Lund University

Andreasen MM (2011) 45 Years with design methodology. J Eng Des 22:293–332.

Andreasen MM, Hansen CT, Cash P (2015) Conceptual Design: Interpretations, Mindset and Models, 1st edn. Springer, Kgs. Lyngby

Baldwin C, Clark K (2000) Design rules: The power of modularity. MIT Press

Blessing LTM, Chakrabarti a (2009) DRM, a Design Research Methodology. Springer

Bonnema G, Veenvliet K, Broenink J (2015) Systems Design and Engineering: Facilitating Multidisciplinary Development Projects. CRC Press

Bucciarelli L (1994) Designing engineers. MIT Press

Buede DM (2012) The Engineering Design of Systems: Models and Methods, 2nd edn. Wiley

Buur J (1990) A Theoretical Approach to Mechatronics Design. Technical University of Denmark

Cabigiosu a., Camuffo a. (2012) Beyond the "Mirroring" Hypothesis: Product Modularity and Interorganizational Relations in the Air Conditioning Industry. Organ Sci 23:686–703. doi: 10.1287/orsc.1110.0655

Chabay RW, Sherwood BA (2011) Matter and Interactions, 3rd edn. Wiley

Chakrabarti A, Blessing LTM (2014) An Anthology of Theories and Models of Design. Springer London, London

Colfer L, Baldwin CY (2010) The Mirroring Hypothesis : Theory , Evidence and Exceptions. Harv. Bus. Rev. 36.

Crawley E, Cameron B, Selva D (2015) Systems Architecture: Strategy and Product Development for Complex Systems. Prentice Hall

Crawley E, Weck O de, Eppinger S (2004) The influence of architecture in engineering systems. MIT esd

Dickerson C, Mavris DN (2010) Architecture and Principles of Systems Engineering. CRC Press, Boca Raton, FL

Duffy A, Andreasen M (1995) Enhancing the evolution of design science. In: Proceedings of ICED. pp 29–35

ECSS (2015) Space Engineering - Interface Management. 1–54.

Eppinger S, Browning T (2012) Design structure matrix - methods and applications. MIT Press

Fernandes J, Henriques E, Silva A, Moss M a. (2014) Requirements change in complex technical systems: an empirical study of root causes. Res Eng Des 26:37–55. doi: 10.1007/s00163-014-0183-7

Grady JO (1994) System integration. CRC press

Harlou U (2006) Developing product families based on architectures - Contribution to a theory of product families. Technical University of Denmark

Haskins C, Forsberg K, Krueger M (2006) Systems Engineering Handbook. INCOSE

Hubka V, Eder WE (1988) Theory of Technical Systems, 2nd edn. Springer-Verlag, New York, NY

Hölttä-Otto K, Chiriac N, Lysy D, Suh E (2014) Architectural Decomposition: The Role of Granularity and Decomposition Viewpoint. Springer

Irwin T (1989) Aristotle's first principles. Clarendon Press

Jarratt TA, Eckert CM, Clarkson PJ, Stacey MK (2004) Providing an Overview during the Design of Complex Products. In: Design Computing and Cognition'04. Springer, pp 239–258

Jørgensen KA (1992) Videnskabelige arbejdsparadigmer (In Danish). Institute for Production, Aalborg Univesity, Denmark

Kapurch SJ (2007) NASA Systems Engineering Handbook. NASA Spec Publ 360.

Kihlander I, Ritzén S (2012) Compatibility before completeness—Identifying intrinsic conflicts in concept decision making for technical systems. Technovation 32:79–89.

Lalli V, Kastner R, Hartt H (1997) Training Manual for Elements of Interface Definition and Control. National Aeronautics and Space Administration, Office of Management, Scientific and Technical Information Program

Levy Y, Ellis TJ (2006) A systems approach to conduct an effective literature review in support of information systems research. Informing Sci Int J an Emerg Transdiscipl 9:181–212.

MacCormack A, Baldwin C, Rusnak J (2012) Exploring the duality between product and organizational architectures: A test of the "mirroring" hypothesis. Res Policy 41:1309–1324. doi:

10.1016/j.respol.2012.04.011

Mahajan S (2014) The Art of Insight in Science and Engineering: Mastering Complexity. MIT Press

Mortensen NH (1999) Design modelling in a Designer's Workbench - Contribution to a Design Language. Technical University of Denmark

Mouzala M (2012) Aristotle's Method of Understanding the First Principles of Natural Things in the Physics I. 1. In: Peitho Examina Antiqua. pp 31–50

OED (2015a) "framework, n.". In: OED Online. Oxford Univ. Press. http://www.oed.com/view/Entry/74161?redirectedFrom=framework. Accessed 19 Nov 2015

OED (2015b) "phenomenon, n.". In: OED Online. Oxford Univ. Press. http://www.oed.com/view/Entry/142352?redirectedFrom=phenomenon. Accessed 19 Nov 2015

Olesen J (1992) Concurrent Development in Manufacturing based on dispositional mechanisms. Technical University of Denmark

Parslov JF, Mortensen NH (2015) Interface definitions in literature: A reality check. Concurr Eng. doi: 10.1177/1063293X15580136

Pedersen K, Emblemsvåg J, Bailey R, et al (2000) Validating design methods and research: the validation square. In: ASME DETC Design Theory and Methodology. ASME, Baltimore, Maryland, p 12

Pimmler TU, Eppinger SD (1994) Integration analysis of product decompositions. In: ASME Design Theory and Methodology Conference Minneapolis, MN. Alfred P. Sloan School of Management, Massachusetts Institute of Technology,

Rahmani K, Thomson V (2012) Ontology based interface design and control methodology for collaborative product development. Comput Des 44:432–444.

Sage AP, Cuppan CD (2001) On the Systems Engineering and Management of Systems of Systems and Federations of Systems. Information-Knowledge-Systems Manag 2:325–345.

Simpson T, Siddique Z, Jiao R (2006) Product platform and product family design: methods and applications. Springer Science and Business Media

Simpson TW, Jiao JR, Siddique Z, Hölttä-Otto K (2014) Advances in Product Family and Product Platform Design. Springer New York

Standard I, ISO/IEC/IEEE (2011) ISO/IEC/IEEE Systems and software engineering -- Architecture description. ISO/IEC/IEEE 420102011(E) (Revision ISO/IEC 420102007 IEEE Std 1471-2000) 2011:1–46. doi:

10.1109/IEEESTD.2011.6129467

Steward D V (1981) Systems analysis and management: structure, strategy and design. PBI (New York)

Suh ES, Weck OL, Chang D (2007) Flexible product platforms: framework and case study. Res Eng Des 18:67–89. doi: 10.1007/s00163-007-0032-z

Suh NP (1990) The principles of design. Oxford University Press New York

Torry-smith JM (2013) Designing Mechatronic Products -Achieving Integration by Means of Modelling Dependencies. Technical University of Denmark

Ulrich K (1995) The role of product architecture in the manufacturing firm. Res Policy 24:419–440.

Ulrich KT, Eppinger SD (2012) Product design and development. McGraw-Hill Higher Education, New York

Weber C (2014) Modelling products and product development based on characteristics and properties. In: An Anthology of Theories and Models of Design. Springer London, pp 327–352

Weck OL De, Roos D, Magee C (2011) Engineering systems: Meeting human needs in a complex technological world. MIT Press

Wheatcraft LLS (2010) Everything you wanted to know about interfaces, but were afraid to ask. Compliance Automation

Wie MJ Van, Greer JL, Campbell MI, et al (2001) Interfaces and Product Architecture. In: ASME 2001 International Design Engineering Technical Conferences And Computers and Information in Engineering Conference. pp 1–12

Wikipedia (2016) Physics. https://en.wikipedia.org/wiki/Physics.

Yasseri S (2015) Interface management of subsea field development. Underw Technol 33:41–57. doi: 10.3723/ut.33.041

Yin R (2013) Case study research: Design and methods. SAGE Publications

Zheng C, Duigou J Le, Bricogne M, Eynard B (2016) Multidisciplinary Interface Model for Design of Mechatronic Systems. Comput Ind 76:24–37.

7 Appended Papers

Paper A (Published, recommended for journal publication, honors)

Jensen, T. V., Parslov, J. F., & Mortensen, N. H. (2015, August). Enabling Reuse of Documentation in New Medical Device Development: A Systematic Architecting Approach. In *ASME 2015 IDETC/CIE DTM Conference* (pp. V007T06A024-V007T06A024). American Society of Mechanical Engineers.

Paper B (Published)

Parslov, J. F., & Mortensen, N. H. (2015). Interface definitions in literature: A reality check. *Concurrent Engineering*, 1063293X15580136.

Paper C (Submitted, currently under peer review)

Parslov, J. F., Gerrard, B., & Mortensen, N. H. (2016). Understanding Interactions in Complex Multi-Technological Products – A First Principle, Physics-based Theoretical Framework. *Research in Engineering Design*

Paper D (Submitted, currently under peer review)

Parslov, J. F., Gerrard, B., & Mortensen, N. H. (2016). Defining Interactions and Interfaces in Complex Multi-Technological Products – A Multi-disciplinary, Physics-based Approach. *Research in Engineering Design*

Proceedings of the ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2015
August 2-5, 2015, Boston, Massachusetts, USA

DETC2015-47272

ENABLING REUSE OF DOCUMENTATION IN NEW MEDICAL DEVICE DEVELOPMENT: A SYSTEMATIC ARCHITECTING APPROACH

Troels Victor Jensen
Department of Mechanical Engineering
Technical University of Denmark
Kgs. Lyngby, Denmark

Jakob Filippson Parslov
Department for New Product Development
Radiometer Medical ApS
Brønshøj, Denmark

Niels Henrik Mortensen
Department of Mechanical Engineering
Technical University of Denmark
Kgs. Lyngby, Denmark

ABSTRACT

Medical device companies are continuously challenged with the ability to prove compliance with increasingly complex regulatory frameworks. Operating under heavy regulatory requirements may therefore cause significant delays to the lead time of new medical devices and thus contribute significantly to time-to-market for even simple medical device development projects. In this paper we illustrate how medical device companies can reduce their research and development (R&D) efforts needed to prove compliance when developing new product families by means of platforming and modularization. The results presented in this paper are based on a two-year empirical case study of a European manufacturer of arterial blood gas (ABG) sampling devices. The core contribution of this paper is a systematic architecting approach that applies the concept of a delta-multi-domain matrix (ΔMDM) to support companies in justifying the reuse of verification and validation (V&V) test documentation packages across new product family designs. The paper introduces an approach to aligning product and documentation architectures by architecture mirroring, and emphasizes the need for having a one-to-one mapping between the product and V&V test view. This will allow for V&V-related documentation to follow the product platform, and thereby enable carry-over of test documentation packages from one product family to another. Hence, this approach can provide significant competitive advantages to companies as it increases R&D efficiency while reducing time-to-market for new medical device development.

INTRODUCTION

Medical device companies are constantly under pressure to prove compliance with regulatory requirements when bringing new medical devices to market or to maintain market clearance for devices already on market. Tightening regulatory requirements demands medical device companies to have increasingly extensive documentation processes in place – processes which are often disproportionately costly compared to the engineering design effort needed to develop the product. The regulatory requirements thus contribute significantly to R&D resource consumption and time-to-market for even simple medical devices. This paper presents a seven step systematic architecting approach conceived during a two-year empirical case study conducted at Radiometer Medical ApS, a globally leading provider of high technologically advanced acute care solutions, which aims at simplifying and automating the phases of acute blood gas testing.

The core contribution of this paper adds to the body of knowledge of modularization and platforming. We specifically show how the need for future reuse of V&V test

 Copyright © 2015 by ASME

documentation across new product family designs can be addressed in the architecting phase. It illustrates how the investment put into new product verification and validation (V&V) testing and documentation can be reused or shared across future product family designs in order to increase R&D efficiency while shortening the product lead time.

In the validation of the proposed approach, emphasis is put into detailing the verification test setup, hence same methodological principles will apply for the validation test setup and the reuse hereof.

The first section introduces related research followed by a step-by-step explanation of the proposed approach, after which the proposed approach is validated based on a real-life case study. At last, a discussion on the used research methods and the obtained results is given followed by a conclusion.

RELATED WORK

Product modeling

In engineering design theory, products can be considered as technical systems of elements and relations [1]. The appealing thing about systems is the recursive nature, meaning that a system-of-interest is composed of subsystems, but can itself also be considered as a subsystem of an even larger system [2]. Abstracting from underlying subsystems by only focusing on the inputs and outputs makes it possible to manage highly complex systems.

Complex technical systems are generally thought to possess both structure and behavior and may exhibit behavioral properties that no subset of their elements has [3]. It is therefore common to model products from both a functional and a physical perspective.

Andreasen [4] proposed a classification of a system's attributes; a system has structural characteristics and behavioral properties. The structural characteristics of the elements are of the kind form, dimension, material, surface quality, and state [5]. In addition to this, the properties are classified as mainly functional (i.e. transformation) and functional properties (e.g. reliability, accuracy, readability etc.) [5]. Other prominent researchers within the engineering design research community such as Suh [6] (i.e. Axiomatic design, AD) and Weber [7] (i.e. characteristics-properties modeling, CPM) has characterized a system's attributes which are fairly similar. In this paper we adopt the definition by [4] for describing a product's structure and behavior.

The following model (Figure 1) is inspired by [4] and represents the authors' understanding of how functions have properties that are realized by structural characteristics belonging to certain components. The property models are partial view sets of the product structure and behavior. Only characteristics that contribute to a certain property is included in a property model.

Product platform and architecture

Product architecture can be defined as the scheme by which the function of a product is allocated to physical components

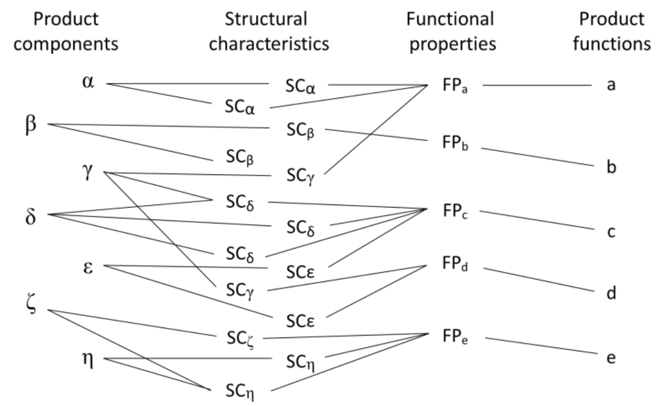


Figure 1: Breakdown-pattern of a system's attributes inspired by Andreasen's work [4].

meaning in detail (1) the arrangement of functional elements, (2) the allocation of functional elements to physical components, and (3) the specification of interfaces between interacting physical components [8]. A company's ability to create product variety resides with the architecture of the product. There are fundamentally two aspects of product variety: the functional variety which is directed towards the marketplace and aims at satisfying diverse customer needs, and the technical variety directed towards the cooperative operations and aims at reducing manufacturing cost [9].

An approach called Design for Variety (DfV) has been proposed where modular product architectures are developed using Quality Function Deployment (QFD), generational variety index (GVI) and coupling index (CI) [10]. The key contribution of this approach is the ability to target areas in the product structure that should be decoupled in order to support the release of future product variants. However, the authors do not discuss the opportunities of decoupling the component with respect to its verification and validation.

The definition of a product family architecture can be separated into three parts: the "common base" that share components within a product family, the "differentiation enabler" that makes variants different from one another and the "configuration mechanism" that defines the rules for variant derivation [11]. Whereas the architecture is a collection of both common and unique elements enabling product commonality and variety, the product platform is a collection of assets shared across a family of products [12], and usually over product generations. Hence, the design knowledge is leveraged from one product to another thus reducing the initial cost. Defining common components is therefore key to the process of identifying a product platform.

The concept of modularization is an effective mean to enable reuse or upgrade of existing functions and features across multiple product variants and families. Reusing or sharing modules not only involves reusing the drawings and specifications but has to do with reusing the investment put into developing, testing and verifying the module [9]. A module is designed so that it captures certain functionality but is described by its structural characteristics. Modules are

 Copyright © 2015 by ASME

derived based on certain drivers. Twelve module drivers has been identified [13], one of which, the “carry-over” module driver, which will be used as the driver for module identification in this paper.
A modular architecture is characterized by modules having few and weak inter-module interactions, but may possess many strong intra-module interactions [8]. Modular architectures are commonly considered as less complex than integral architectures, however it may be the case that increased modularity leads to increased topological complexity due to the idea that complexity can be distributed differently across the architecture [14]. Full modularization is reached when geometry, energy, material or signal of one component can be changed in order to fulfill a certain functional requirement supporting a customer need without requiring other components to be changed [10]. Hence, an important part of modularization is defining the interactions and interfaces between the modules [15]. Often times, specific interface designs ‘survive longer’ than the components they connect when looking across generations of products [16]. Various researchers have defined different types of interfaces at an architectural level, e.g. [17] suggests four types of interactions among elements in the architecture: material, energy, information, and spatial. Another contribution consolidates several research efforts into a functional basis [18], however the important types of interactions will be context specific and vary from product to product [19].

Visual modeling tools
There are fundamentally three different types of visual modeling tools for modeling systems; matrix-based, network-based, and block diagram-based. A literature survey finds the design structure matrix (DSM) to be one of the most used methods in engineering design research dealing with modularization [20]. Applying the DSM method in product decomposition involves three steps: 1) decomposition of the product into smaller elements, 2) documentation of interactions between the elements, and finally 3) clustering the elements into chunks [21]. Clustering techniques are widely used within matrix-based modeling for module identification [22]. Another tool called multi-domain-matrix (MDM) covers both DSMs and their relations – the domain mapping matrix (DMM) [23]. The MDM technique is adopted and used at the core of this paper.

Medical device verification and validation
The medical device and drug regulatory frameworks are extremely complex and can significantly delay manufacturers attempting to bring new devices and drugs to market [24]. Medical device manufactures are responsible for classifying their products according to the guidelines set by public authorities and are required to make a premarket notification prior to launching their products. Three classes exist according to identified patient risk ranging from low-risk products such as non-sterile gloves to artificial hips and spinal fixation systems ranked highest. Medical devices posing a certain degree of risk must obtain market approval through a 510(k) process or a premarket approval application [25], [26]. In order to obtain market approval, engineering products must demonstrate performance for their intended use before they are released to market. This process of obtaining market approval requires thorough verification and validation of the medical device. The validation process ensures that the device meets the purpose throughout the development process by demonstrating the consistency and completeness of the design with respect to the initial ideas of what the product should be used for and how, hence the mapping from use activities to product functions [27]. The process of validating the medical device is concerned with ensuring that, as the design and implementation develops, the design output (i.e. specifications) from each development phase fulfills the design input (i.e. requirements) that was output from the previous phase. In general, verification is a process that occurs as part of the activities device design, process design, and production development [28]. Design verification is an evaluation activity that involves comparison of design outputs (the specification or outcome of design-related activities such as drawings, risk analysis and test results) with design inputs (the requirements set for the design). These comparisons involves a range of methods like simulation tools, analysis, testing etc. in order to provide the required evidence that the product specification or outcome of the design is equivalent to the requirements. In relation to the product modeling section one could argue that what is being verified is whether the configuration of certain component and their characteristics (i.e. documented in design specifications) exhibits the intended functional properties (i.e. documented in requirements). Validation however, is concerned with ensuring the completeness of requirements in relation to customer needs or activities. These processes require extensive documentation proving compliance with given regulations, which affect medical manufactures’ ability to bring new devices and drugs to market.
Some specific methods for medical device development have been proposed in literature. A four step method for developing modular product architectures and accessing the optimal number of modules in a medical device has been proposed [29], and a product design process model specifically tailored for medical devices has been introduced [30]. The intent is to present the fundamental information that designers should understand when initiating the development of medical devices.

PROPOSED APPROACH

This section presents the proposed approach (Figure 2) in a step-by-step guide. The approach is applicable to companies who seek to analyze existing product families and/or to develop new product families derived from an existing one.
Step 1–3 in the approach involve gaining insight into the customer use activities, product functions and the physical elements of the products.

Copyright © 2015 by ASME

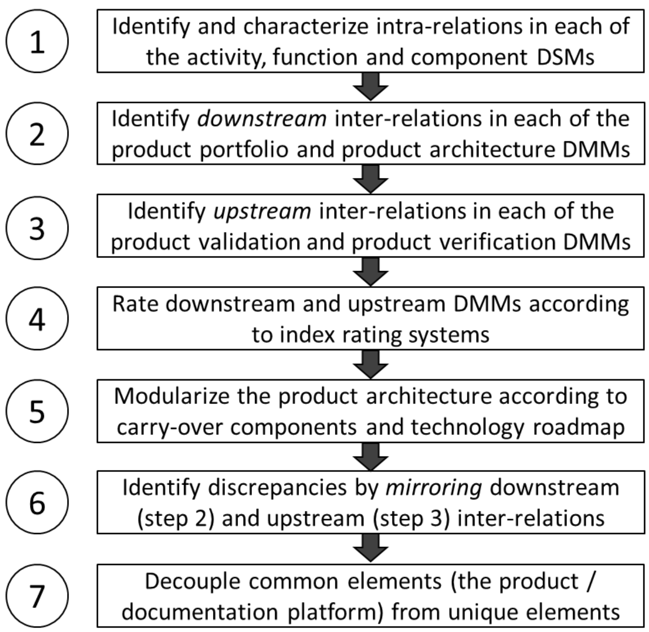


Figure 2: Step-by-step outline of the proposed approach.

Step 4 involves quantification of dependencies across the three domains, namely downstream and upstream relations. Step 5 and 6 concern the analysis, modularization and identification of potentials for test reuse, and finally step 7 focuses on the decoupling of the identified dependencies between common and unique elements.

The approach requires certain preconditions that must be met prior to applying the approach in practice. First of all, it presupposes that the company has:

1) A clear product and technology roadmap of the product families on a concrete modular level.
2) Domain matter insight into own products in order to take any qualified actions.
3) A test plan covering the functional properties of the product program and its families.

The nature of these steps will differ depending on whether an existing product family is being analyzed, and the data is readily available or if the family is being synthesized from scratch. In the latter case the process may take on a highly iterative nature.

The multi-domain modelling (MDM) tool [23] has been adapted to represent the approach, as it displays relationships between multiple elements in a compact, visual and analytically advantageous format (Figure 3). It contains design structure matrices (DSM) [21] along the diagonal, namely the customer use activities, product functions and product components, and domain mapping matrices (DMM) off-diagonal, on one side representing downstream inter-relations (product portfolio and product architecture), and on the other side upstream inter-relations (verification and validation tests). The core idea is to mirror the DMMs across the diagonal hence aligning the product and V&V test documentation architectures associating the SE V-model [31].

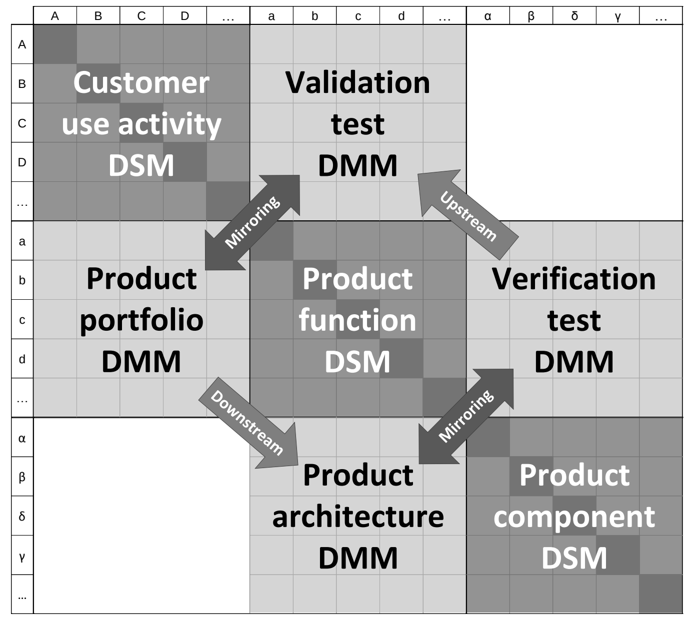


Figure 3: Fundamental principles of the ΔMDM mechanisms supporting the systematic architecting approach.

The purpose of the modelling technique is to provide an overview of intra-relations between the elements of the DSMs and its inter-relations represented in the DMMs. As the multi-domain matrix is used to identify dependencies or deltas across multiple views, we introduce the concept of a delta-multi-domain matrix (ΔMDM) with the advantage of supporting the discrepancy or delta analysis and architecture alignment (step 6). In this paper rows represents *providing* elements and columns represent *depending* elements. In the following, the seven steps of the approach will be explained.

Step 1

Takes on the task to model the customer use activities, the functional break down and the physical structure of the product in the DSMs. This provides an overview of what the product is used for, how it works, and how it is realized. In the function and component DSMs, interactions such as material (M), energy (E), information (I), and spatial (S) arrangement are used when filling out the matrices [17], [18].

Step 2

Involves understanding and identifying the downstream inter-relations between the DSMs – the product view. These intermediate views thus represent the product portfolio and the product architecture view respectively. Firstly, risk sensitive activities are identified based on their probability (P), detection (D), and severity (S), all ranked from 1-10 and multiplied giving the risk priority number [32]:

$$RPN = P \; x \; D \; x \; S$$

 Copyright © 2015 by ASME

This will enable the company to prioritize the subsequent mapping of inter-relations. Firstly, the customer activities are mapped to product function revealing the product portfolio DMM. Next, product functions are mapped to product components revealing the product architecture DMM. In order to fill in these relations, the company must gain insight into the underlying property models connecting structural characteristics (SC_n) such as form, dimension, material, surface quality, and state [5] to functional properties (FP_n).

Step 3
Concerns the mapping of the verification and validation tests. Tests validating the functions against the use activities are filled out in the validation test DMM. The verification view shows what components have been or are planned to be tested in relation to specific functions. Like the product architecture view, the verification view too has a lower level similar to the product architecture, except here, only the components that were planned for or actually tested together are mapped. In principle, verification testing is about physically verifying that the property models conceived during synthesis of the product results in desired product behavior that does not pose a risk to patient or operator safety.

Step 4
Involves rating the product view and the V&V test documentation view according to predefined rating systems. In the product view, it is assumed that different components and their structural characteristics (SC_n) have different sensitivity of impact on a given functional property (FP_n) related to a function. The same is the case for the functional properties, as these have different sensitivity of impact on the customer use activities. Thus rating (Table 1) the relations according to sensitivity of impact will expose those elements (function and/or components) that have a high impact on the (activities and/or functions) when changed. These elements may be important to protect or understand in detail.

Table 1: System Impact Index (SII) - Rating system for assessing sensitivity of magimpact of a component design change to the system's functional properties and customer use activities respectively.

Index	Description
9	Eliminates the functional property / activity
6	Causes major changes to the functional properties / activities, and other parts *will* be affected
3	Causes a minor change to the functional properties / activities, and other parts *may* be affected
1	Causes a minor change to the functional properties / activities. Other parts *will not* be affected
0	No impact on functional properties / activities

In the verification and validation test view, each function (functional properties) or activity (procedural step) may have been documented in several tests with different configurations of components or functional properties. These tests vary according to level of resources required [man-hours / test] and time-consumption [hours]. The rating system (Table 2) thus evaluates both the number and magnitude of the component-to-function (verification) and function-to-activity (validation) tests.

Table 2: Test Impact Index (TII) - Rating system for assessing the extent of product verification and validation testing affected by component design changes.

Index	Description
9	Part of several both major and minor tests (resources and time)
6	Part of few major and several minor tests
3	Part of several minor verification / validation tests
1	Part of few minor verification / validation tests
0	Not part of verification / validation test

Step 5
Concerns sequencing and clustering the DSMs. The activity DSM is sequenced according to the order by which the activities take place. The product function and product component DSMs are clustered according to their dependencies. In the component DSM common and unique components are identified across the product families based on the predefined product and technology roadmap. Clustering the components thus exposes those tests that span across common and unique components. These tests cannot be reused directly across different product variants; therefore these should be minimized if possible. Clustering should be applied with the purpose of identifying 'carry-over' elements, which constitutes the platform.

Step 6
Involves mirroring the product view in the V&V test documentation view using the ΔMDM so as to align the product and V&V architectures. Next, the apparent deltas between the downstream and upstream views as conceived during the design of the products and the test setup is identified. Identified delta-relations are unfolded in a detailed ΔMDM model for the identification of dependencies with the potential for decoupling creating a one-to-one mapping between the two views. Elimination of these deltas should be pursued so that each property model is documented by a verification test only taking into account the components included in the property model.

Having completed step 1–6 there are specifically two classes of components which should attract attention:

1) Those common or unique components that seem to have high impact on the functions (functional properties) if changed and in the same time take part in many and comprehensive tests. If these functions in addition are critical to patient or operator safety (following the RPN value) we may want to protect or at least fully understand these critical components.

 Copyright © 2015 by ASME

2) Those unique components that only have minor impact on a function (functional property) if changed that spans common and unique components, but which are extensively tested in the V&V view. There may be a potential to scope these components out of the tests by intensively scrutinizing the relation and documenting it properly and if necessary, make a design change to minimize or eliminate the impact completely.

Step 7

Is concerned with decoupling of verification tests that span common and unique elements. As mentioned earlier, verification tests are intended to prove the correctness of the conceived property models. In order to decouple verification tests one must first understand the design on a property model level and if necessary decouple the property models through design changes. Decoupling can be achieved in two ways:

1) Perform explorative testing and gain ample insight into the behavior (output) of the platform by *varying* inputs.
2) Apply design changes aiming at manipulating the property models to eliminate dependencies.

Finally, complete interface and interaction requirements to the platform are documented based on input from the property models with unambiguous acceptance criteria. The interface and interaction requirements metaphorically represent the "fence" protecting the platform from outside changes. The justification for reusing a test thus relies on proving whether the unique component complies with this set of requirements.

Validation testing can generally only be done on the final produced product. Validation testing is intended to prove the completeness of the functional requirements in relation to customer needs or activities. Given that certain functional requirements can be ascribed to certain areas of the system, it is possible to ascribe customer needs to a platform and let validation tests follow the platform.

CASE STUDY: VALIDATION OF THE APPROACH

The presented approach has been validated through a two-year empirical case study conducted in a medical device company, Radiometer Medical ApS. Throughout the study, both qualitative and quantitative data has been gathered from divergent sources through interviews of domain experts in the case company and by reviewing codified information from multiple product data and documentation sources respectively.

In order to maximize patient safety, the company has chosen to expand its core business, blood gas analyzers, to include the blood sampling process by introducing first one family of blood gas sampling devices followed by a more advanced one (Figure 4). The core function of the blood gas sampler is to retain the properties of the patient's blood after extraction and during transport to the analyzer ensuring reliable and accurate measurement results.

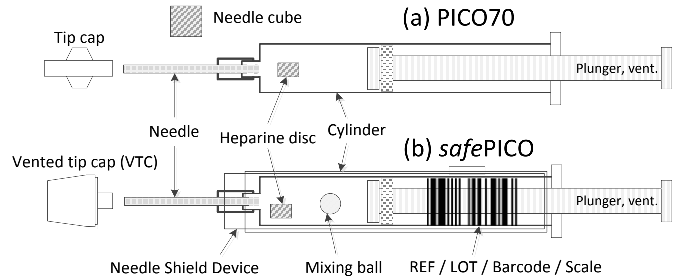


Figure 4: Schematic representations of the two ABG sampler families: (a) PICO70 and (b) *safe*PICO, illustrating main product sub-systems and components.

The results presented in this paper represent a historical analysis of two arterial blood gas (ABG) sampling device families currently on the market: the PICO70, a self-filling ABG sampler (Figure 4, a), which was first introduced in the 1990's, and the advanced *safe*PICO (Figure 4, b), which was introduced 5 years later, with additional features such as a vented tip cap (VTC), intended to support the user in easily expelling air from the sampler while encapsulating the blood specimen in a closed blood system. Another feature of the *safe*PICO is the automated mixing of the blood sample when used together with Radiometer's blood gas analyzers. The mixing process supports an even distribution of the heparin in the sample, thus minimizing the risk of blood coagulation. Further, it features a needle shield device (NSD) for one-hand disposal of the needle and a barcode for securing correct match between the sampler, the patient, and the results of the blood gas analysis. The continuous product improvements aim at increasing patient and operator health and safety by reducing risks related to device handling.

This particular area of business, the ABG sampling devices, has been chosen for this case based on its characteristics, which are generally representative to those challenges confronting the medical device industry, namely the ability to manage the product complexity driven by enhanced customer needs while regulatory requirements for receiving market approval continuously are tightened, increasing the lead time for bringing new medical devices to market.

In the following, the proposed approach will be validated by applying the approach to the above mentioned ABG sampler case.

Step 1: Understanding the use activity and the product

When developing medical devices it is essential to understand the use activities of the devices in detail, meaning mapping the activities that the devices undergo. The activities of the blood sampling process have been conceived through qualitative interviews with employees at the company and the results can be seen in the customer use activity DSM (Figure 8). Having understood what activities the samplers are used in, their product architecture was further analyzed.

A Radial Product Architecture (RPA) model has been developed and used to make the communication with the domain experts more clear and to verify the correctness of the

Copyright © 2015 by ASME

data (Figure 5). The RPA mapping shows how functions are realized by different components, how functions interact by means of material (M), energy (E), information (I), and spatial (S) arrangement, and how each of the two sampler families are structurally composed.

The complex pattern of product inter-dependencies in-between the functional and physical domain as well as intra-dependencies within the functional domain shown in the RPA tool drives the product complexity. It was found that the reason for the complexity of these fairly simple products is due to the amorphous blood medium contained in the sampler which integrates the functionality in a complex mapping to the part structure, e.g. the blood acts as a force transmitting medium and thereby functionally integrates the components that are in contact with the blood. Blood consists of many different elements (i.e. blood cells, plasma etc.), which if treated incorrectly may rupture and cause the blood properties to change. The inherent unstable nature of blood thus adds another layer of complexity to the product architecture and challenges the company's ability to justify that e.g. added functionality or modified product component designs neither affect performance nor the usability of the blood gas sampling device.

By applying the RPA (Figure 5), we demonstrate how adding a component like the mixing ball, when transitioning from PICO70 to *safe*PICO, has a high impact on the rest of the product. The mixing ball contributes to 7 different functions by means of its different characteristics, e.g. it influences the ability to prevent blood coagulation by applying mechanical work on the blood sample. This causes advection (i.e. flow of material with a conserved property) of the heparinized blood out into regions of the blood sample where there is a low concentration of heparin causing diffusion. In addition, it also interacts with the environment, through a magnetic field from an external mixer unit which forces the mixing ball to follow a linear path inside the cylinder thus mixing the blood.

A key point from the RPA mapping is the fact that several functional properties (FP_n) are realized across both common and unique components and their structural characteristics (SC_n). For this reason testing and verification has been done on a complete product level and thus had to be redone every time a unique component had been introduced. Thus one could not justify the effects of an added component on the associated functional property. Another point is the fact that the cylinder, plunger, and heparin disc are all shared across the two product families.

Step 2: Filling out the product portfolio and architecture DMMs

The product architecture analysis (using RPA mapping) has provided the necessary input to fill out the function and component DSMs as well as the product architecture DMM

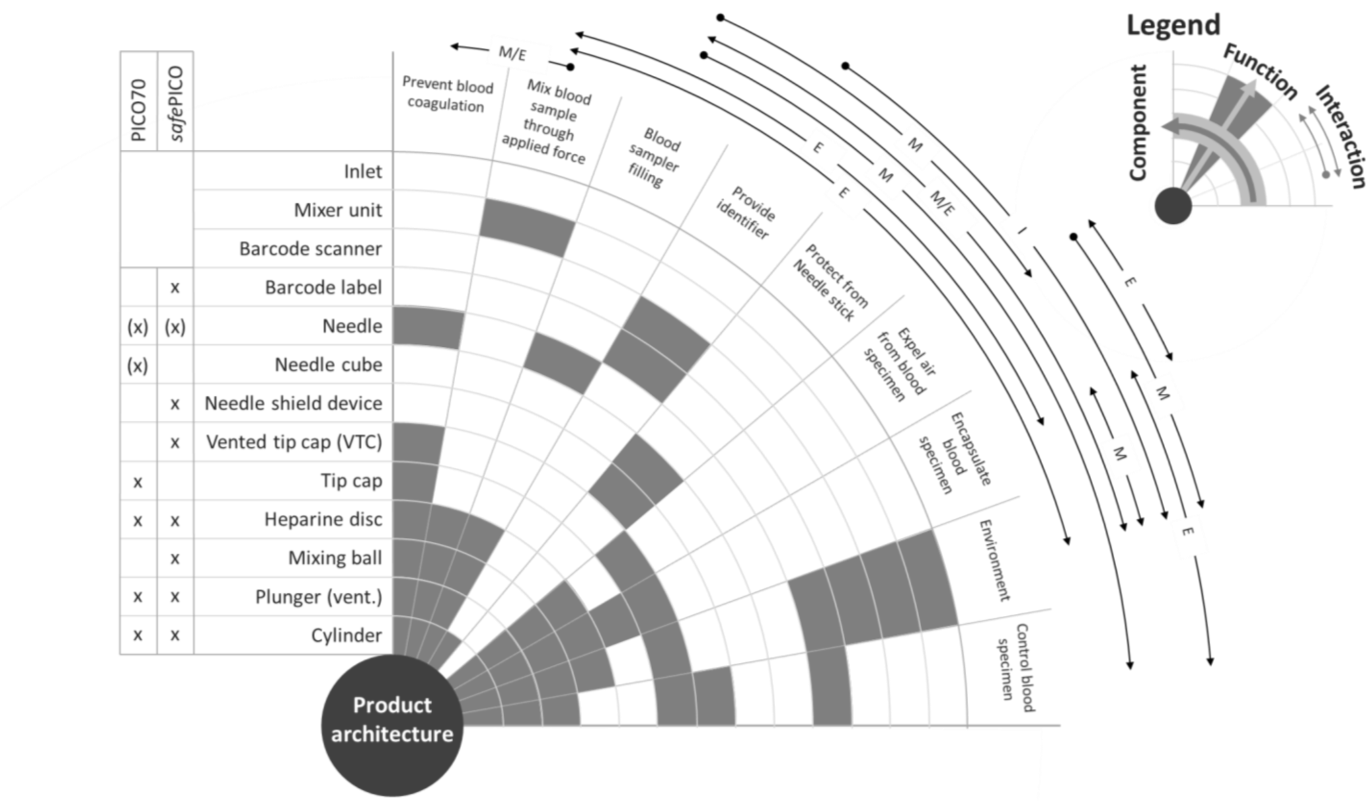


Figure 5: The radial product architecture (RPA) mapping illustrating the composition of the two arterial blood gas (ABG) sampler device families.

Copyright © 2015 by ASME

(Figure 8). What should be noted in relation to the product architecture matrix is the fact that every cell is based on an understanding of an underlying property model. These property models may not necessarily be materialized in a document but rather rest in the minds of the domain experts. This is why this particular view as well as the mapping from activity to function (i.e. product portfolio) has been filled out in collaboration with domain experts. As one cannot unfold all property models, the risk priority number (RPN) was applied so as to focus the resources on those inter-relations that support activities associated with high risk to patient and operator safety. This is in particular recommended for large systems analysis – the RPN helps a company to focus its resources on the most critical parameters of the system.

Step 3: Filling out the verification and validation test DMMs

Having filled out the product architecture view in step 2, the task in step 3 is to understand and map out the components being tested together in relation to specific functions. Again, the verification test matrix must be understood on a lower level to be filled out properly; what properties related to what functions were tested? And what component characteristics were included in the respective tests? This level of detail seems too complex to be visualized at the top layer of the DMM (Figure 8). One should therefore at this step, only note whether a cell is filled out. This means that a given component was tested up against a property related to a given function.

Step 4: Rating the product and V&V test views

Up until now the DMMs representing the downstream and upstream inter-relations on the top layer only show a binary system. In order to be able to gain additional insight and to pinpoint critical elements in relations to V&V test reuse, step 4 involves quantification of both the product view and the V&V test view.

By using the System Impact Index (SII) to rate the impact of components design changes to the system's functional properties (Table 1), it is found that the cylinder and plunger are both highly functionally integrated and have high probability of affecting the system's functional properties if changed (Figure 6).

In addition, by applying the Test Impact Index (TII) in Table 2 to the V&V test view (Figure 7), it was found that the functionally integrated components (i.e. cylinder, plunger, and heparin disc) are also the ones represented in most tests (x-axis) which require heavy resources (y-axis), and time consumption (bubble size). Because of this, and since these components are all common across the families, these will be a part of the platform and should be protected against changes.

The mixing ball is the second most resource- and time-consuming unique component from a verification test point of view. To understand why, one would have to decompose further to expose the distribution of resources per test, which we elaborate on in step 6.

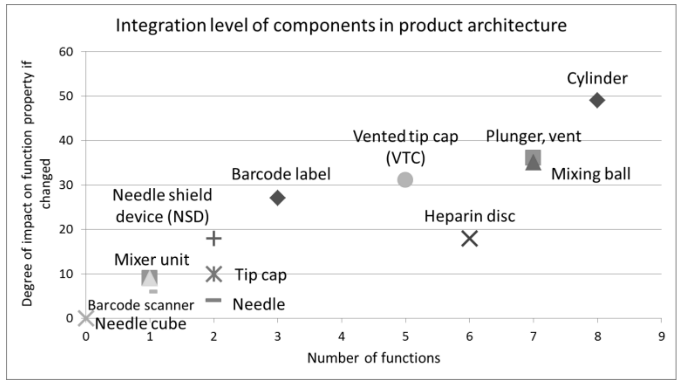


Figure 6: Integration level of components vs. degree of impact in the functional domain.

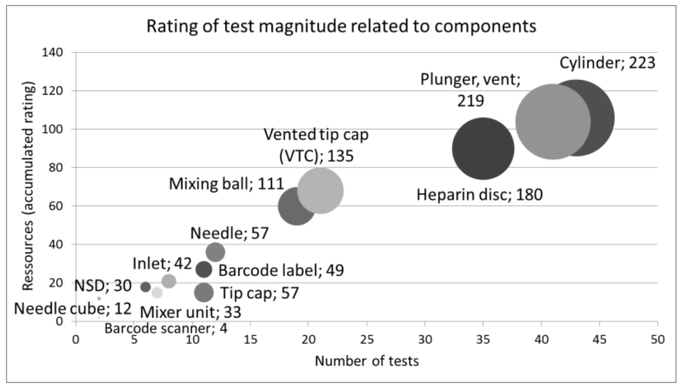


Figure 7: Test impact index of component vs. accumulated resource consumption.

Step 5: Clustering for carry-over of common components

Here, the customer use activity DSM is sequenced (time-based) and the product function and component DSMs are clustered (static) according to its intra-dependencies. Based on the product and technology roadmap, the identified common elements (with assistance from the RPA model) are clustered across the two product families: PICO70 and *safe*PICO (Figure 8). These components will represent the platform which is to be carried over from one family to another.

Earlier, we found the cylinder, plunger and heparin disc to represent common components, thus these have been clustered. The MDM shows, at a top-layer the amount of tests spanning several both common and unique components in the same test thus making the tests product family-specific. Ideally, the company should strive for a one-to-one mapping between the product view and the V&V tests pertaining solely to the platform. This will allow for tests to follow the platform without any changes or rework needed.

Step 6: Mirroring the product architecture into the verification test DMM

Applying the mirroring technique in the ΔMDM model to the holistic architecture approach reveals the delta between

 Copyright © 2015 by ASME

relations across product and V&V test documentation views (Figure 8).

By mirroring the architectures, it is found that more components are included in multiple verification tests regardless of their influence on the given function of the product. The same has been revealed in the function-to-activity validation tests, where functions have been included in the test although these do not support the given use activity, and vice versa. These deltas drive complexity when test documentation is to be reused across future variants or families – e.g. if a new unique component is to be added or an existing component is to be redesigned in order to enable new product functionality or enhance an existing one, the company cannot exchange these physical elements as they are 'locked in' from a test perspective. Here are a couple of findings:

1) From product architecture view, the vented tip cap (μ) supports (rated 3) the function *control blood specimen volume* (i), but is not accounted for in the verification test view. A vented tip cap captures a small amount of "dead" blood volume which affects users' ability to accurately control blood sample volume. A mitigation of moving the scale slightly in production was implemented thus avoiding retesting.
2) A test verifying the effects of a plastic barcode label (ν) on oxygen diffusion (g) were performed (rated 1); a relation which was not supported in the product architecture view. This means that the test was performed even though it may not have been necessary.

Aside from the discrepancies disclosed during the mirroring exercise at a top-layer, the critical property models and its related tests spanning across common and unique components must be unfolded (Figure 9). Given that not all components and their structural characteristics affect a given functional property equally, it is of interest to highlight those tests where

Sampler product program modeling		CUSTOMER USE ACTIVITY									PRODUCT FUNCTION									PRODUCT COMPONENT												
		H	B	E	F	D	A	G	C	I	i	h	c	d	e	a	b	f	g	κ	ι	ε	δ	α	μ	ζ	η	θ	β	ν	γ	λ
Pair sampler to patient by scanning barcodes	H													3																		
Set plunger to desired value	B	X									9	9																				
Draw blood from patient	E		X									9	9																			
Shield the needle	F											9			1																	
Discard the needle	D				X							9			6																	
Press plunger to ventilate sampler	A					X		X				9						3														
Seal the sampler	G					X	X												9													
Mix blood specimen	C						X	X																								
Transport sampler to analyzer	I								X			9																				
Control blood specimen volume	i		6	9			1						M						E	6	6	1	1									
Environment	h	6							6	6			M/E	I			E	M	M	1	1	1	1	1	1		1	1		1		1
Blood sampler filling	c		3	9								M/E								6	6	6	1		1		1	6		1		
Provide identifier	d	9																		1										1	1	
Protect from needle stick	e				9	9														3	3	3	1				1	3	1			
Prevent blood coagulation	a								3								M/E			6	6	6	3	1	3	1	1	1	1	1		
Mix blood sample through applied force	b								9			~~E~~							E	6	6	6	6	6	6		1	1		3		6
Expel air from blood specimen	f						6	6					M						E	6	6	3	3		6							
Encapsulate blood specimen	g		1				3	9			~~E~~	~~M~~					E	E		9	9	3	6		9	9				1		3
Cylinder	κ										9	9	6	1		3	6	6	9		E	E	E		E	E		E		E		
Plunger, vent	ι										9	1	9			1	1	6	9	M/E		E	E		E							
Heparin disc	ε											1	3			9	1	3	1	E	E		E									
Mixing ball	δ										1	9	6			6	9	3	1	E	E	M/E		E								
Mixer unit	α											9				9							E		E							
Vented tip cap (VTC)	μ										3	9				1		9	9	E			E	E								
Tip cap	ζ															1			9	E												
Needle shield device (NSD)	η										9				9													E		E		
Needle	θ												3			1				E							E		E			
Needle cube	β														6													E				
Barcode label	ν										9	9		9						E							E				E	
Barcode scanner	γ											9		9																E		
Inlet	λ											9																				

Identified platform (common elements)

Figure 8: Product and test documentation architecture overview using the principles of ΔMDM: Identification of critical systems elements, platform identification, architecture mirroring, and discrepancy analysis.

Copyright © 2015 by ASME

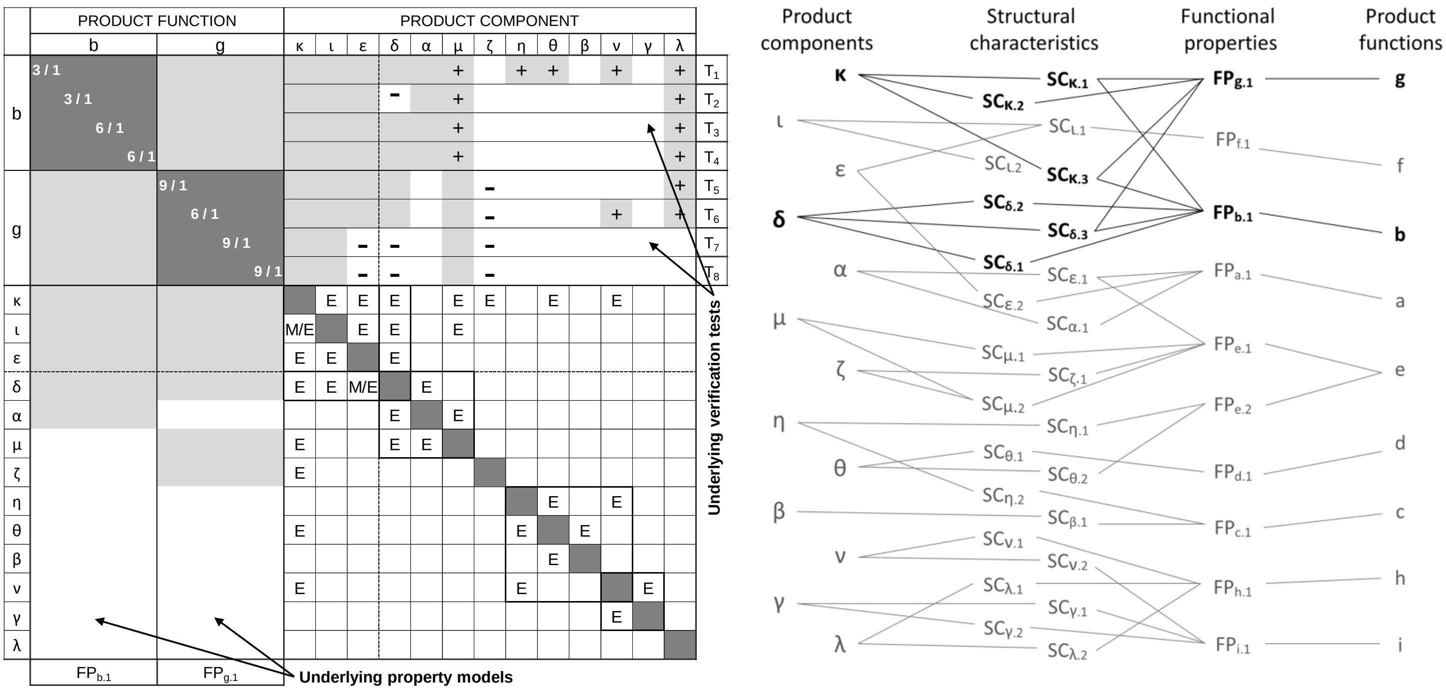


Figure 9: Unfolded mirror of the ΔMDM view showing product and test documentation views (left), and a network diagram (right) highlighting the two functional properties ($FP_{g.1}$ and $FP_{b.1}$), both figures illustrating the property models spanning across the identified platform.

both unique and common components take part and where the unique components' level of impact on the property is low. One such example could be the mixing ball (δ). According to the product architecture (Figure 8) the mixing ball has a strong impact (rated 9) on the *effective mixing property* ($FP_{b.1}$) related to the function *mix blood sample through applied force* (b) and a minor impact (rated 1) on the *gas diffusion property* ($FP_{g.1}$) related to the function *encapsulate blood specimen* (g). The unfolded ΔMDM view (Figure 9, left) shows a sample of those deltas identified during the top-layer discrepancy analysis. The model shows how multiple verification tests either include components that are not part of the property models (+), or components that are part of the property models but have not been included in the tests (-). Further, inter-relations in the component-to-function are identified (Figure 9, right). These views reveals the deltas and hence the potential for decoupling achieving a one-to-one mapping between the product and documentation architecture.

By analyzing the identified discrepancies (Figure 9) we expose to what extent each of the mixing ball's (δ) characteristics contribute to the given property (Figure 10). For example its material ($SC_{\delta.1}$; rated 9), form ($SC_{\delta.2}$; rated 9), and surface quality ($SC_{\delta.3}$; rated 1) all affect $FP_{b.1}$. However $SC_{\delta.3}$ of the mixing ball also has a minor effect on $FP_{g.1}$ (rated 1). When analyzing the property model for $FP_{g.1}$, it is primarily the characteristics of the cylinder (κ) that contributes: material ($SC_{K.1}$; rated 9), surface quality ($SC_{K.2}$; rated 9) and form ($SC_{K.3}$; rated 6). Step 7 will address how to decouple these dependencies.

3 / 1		T_1	b	Product functions
3 / 1		T_2		
6 / 1		T_3		
6 / 1		T_4		
	9 / 1	T_5	g	
	6 / 1	T_6		
	9 / 1	T_7		
	9 / 1	T_8		
$FP_{b.1}$	$FP_{g.1}$	Functional properties		
	9	$SC_{K.1}$	κ	Product components
	9	$SC_{K.2}$		
$\boxed{1}$	6	$SC_{K.3}$		
9		$SC_{\delta.1}$	δ	
9		$SC_{\delta.1}$		
1	1	$SC_{\delta.3}$		

Figure 10: Matrix showing two property models – a complex mapping between tests, product functions, functional properties, structural characteristics, and product components.

Step 7 - Decoupling the property models

As mentioned in the description of the approach, there are two consecutive possibilities to decouple the mixing ball (δ) from the *gas diffusion property* model ($FP_{g.1}$):

1) To make explorative testing where $FP_{g.1}$ is tested with mixing balls with varying surface qualities to understand the impact of variation. An acceptable range of surface qualities could then be defined which then guaranties properties of the platform and protects it from these outside variations.

Copyright © 2015 by ASME

2) If necessary, to make a design change by improving the surface quality or material of the mixing ball, in which case the impact of the mixing ball would be minimized or eliminated from the *gas diffusion property* ($FP_{g.1}$) model.

As Figure 10 reveals, two characteristics ($SC_{\delta.3}$; rated 1 and $SC_{K.3}$; rated 1) span across the two property models ($FP_{g.1}$ and $FP_{b.1}$). The structural characteristic $SC_{K.1}$ is a characteristic of the cylinder (marked with a square), while $SC_{\delta.3}$ is a characteristic of the mixing ball (marked with a circle). In this case, choosing step 2 may be more effective in decoupling the property models however the cost price of the mixing ball might go up as a consequence of improving the surface quality. One must carefully evaluate these trade-offs before making any definite decision.

Through application of the systematic seven step architecting approach, companies gain deep product insight thus providing them with sound rationales to justify the reuse of test documentation across new product family designs.

DISCUSSION

Modeling complex systems in a way that easily reads and communicates certain selected aspects about the system to relevant stakeholders is not an easy task. In this paper, we proposed the usage of a radial product architecture (RPA) model (step 1) to meet these challenges. Applying more visualization tools during information gathering might raise the data validity, as the reliability of domain experts' statements are confronted by applying both logical and visual modeling tools. Thus, we believe that by applying these, misconceptions will be minimized. The limitations of the RPA model though, has to do with the level of complexity it can represent. A suitable granularity level should always be sought. However for the purpose of mirroring (step 6) and further data processing we introduced the MDM-based delta-multi-domain matrix (ΔMDM). The MDM model was adopted as we found it to be a more useful tool for mirroring thus letting the RPA model ensure reliable input to the ΔMDM.
Another point of discussion is the feasibility associated with creating a complete mapping of the system. The modeling of property models, may for example never reach a "complete" representation of the system, because they denote partial, fragmented views of the system driven by a certain purpose. The virtue in this approach is rather to identify which of the product attributes that are critical – either from a health & safety, platforming, performance, or market etc. point of view.

CONCLUSION

In this paper, we have shown how the proposed seven step architecting approach can support companies in lowering the documentation burden related to bringing new medical devices to market. The approach builds on the principles of platforming and modularization enabling reuse of tests and test documentation across multiple product variants. By applying the ΔMDM modeling technique, it is possible for medical device manufacturers to create a solid decision basis for reuse of verification and validation tests documentation. The results of the validation case study showed how the approach enables identification of specific areas within the product architecture that purposefully could be decoupled in order to improve the likelihood of reusing tests across product families. The approach emphasizes that understanding the underlying property models governing the product architecture are key to enabling reuse of test documentation, in that it provides the necessary insight for creating sound rationales needed to justify the reuse of test documentation.

AKNOWLEDGEMENTS

We would like to thank Radiometer Medical ApS for providing valuable insight into the case study presented in this paper, and all the people involved in the research.

REFERENCES

[1] Hubka, V., & Eder, W. E. (1988). Theory of technical systems: a total concept theory for engineering design. *Berlin and New York, Springer-Verlag, 1988, 291 p., 1*.
[2] Ulrich, K. T., Eppinger, S. D., & Goyal, A. (2011). Product design and development (Vol. 2). New York: McGraw-Hill.
[3] Crawley, E., de Weck, O., Eppinger, S., Magee, C., Moses, J., Seering, W., & Whitney, D. (2004). The influence of architecture in engineering systems. Engineering Systems Monograph, 2006.
[4] Andreasen, M. M., Howard, T. J., & Bruun, H. P. L. (2014). Domain Theory, its models and concepts. In An Anthology of Theories and Models of Design (pp. 173-195). Springer London.
[5] Andreasen, M. M. (2011). 45 Years with design methodology. Journal of Engineering Design, 22(5), 293-332.
[6] Suh, N. P. (1995). Axiomatic design of mechanical systems. Journal of Mechanical Design, 117(B), 2-10.
[7] Weber, C. (2014). Modelling products and product development based on characteristics and properties. In An Anthology of Theories and Models of Design (pp. 327-352). Springer London.
[8] Ulrich, K. (1995). The role of product architecture in the manufacturing firm. Research policy, 24(3), 419-440.
[9] Erens, F., & Verhulst, K. (1997). Architectures for product families. Computers in industry, 33(2), 165-178.
[10] Martin, M. V., & Ishii, K. (2002). Design for variety: developing standardized and modularized product platform architectures. Research in Engineering Design, 13(4), 213-235.
[11] Du, X., Jiao, J., & Tseng, M. M. (2001). Architecture of product family: fundamentals and methodology. Concurrent Engineering, 9(4), 309-325.
[12] Robertson, D., & Ulrich, K. (1998). Planning for product platforms. Sloan management review, 39(4).
[13] Ericsson, A., & Erixon, G. (1999). Controlling design

 Copyright © 2015 by ASME

variants: modular product platforms. Society of Manufacturing Engineers.
[14] Sinha, K., & de Weck, O. L. (2013, August). Structural Complexity Quantification for Engineered Complex Systems and Implications on System Architecture and Design. In ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (pp. V03AT03A044-V03AT03A044). American Society of Mechanical Engineers.
[15] Jiao, J. R., Simpson, T. W., & Siddique, Z. (2007). Product family design and platform-based product development: a state-of-the-art review. Journal of Intelligent Manufacturing, 18(1), 5-29.
[16] Lalli, V. R., Kastner, R. E., & Hartt, H. N. (1997). Training Manual for Elements of Interface Definition and Control. NASA Aeronautics and Space Administration, Lewis Research Center, Cleveland, Ohio.
[17] Pimmler, T. U., & Eppinger, S. D. (1994). Integration analysis of product decompositions (pp. 343-351). Cambridge, MA: Alfred P. Sloan School of Management, Massachusetts Institute of Technology.
[18] Hirtz, J., Stone, R. B., McAdams, D. A., Szykman, S., & Wood, K. L. (2002). A functional basis for engineering design: reconciling and evolving previous efforts. Research in engineering Design, 13(2), 65-82.
[19] Browning, T. R. (2001). Applying the design structure matrix to system decomposition and integration problems: a review and new directions. Engineering Management, IEEE Transactions on, 48(3), 292-306.
[20] Holmqvist, T. K., & Persson, M. L. (2003). Analysis and improvement of product modularization methods: Their ability to deal with complex products. Systems Engineering, 6(3), 195-209.
[21] Steward, D. V. (1981). Systems analysis and management: structure, strategy and design. (New York): PBI.
[22] Eppinger, S. D., & Browning, T. R. (2012). Design structure matrix methods and applications. MIT press.
[23] Maurer, M. S. (2007). Structural awareness in complex product design (Doctoral dissertation, Universität München).
[24] Masterson, F., & Cormican, K. (2013). Overview of the Regulation of Medical Devices and Drugs in the European Union and the United States. Therapeutic Innovation & Regulatory Science, 47(6), 715-722.
[25] Kramer, D. B., Xu, S., & Kesselheim, A. S. (2012). Regulation of medical devices in the United States and European Union. New England journal of medicine, 366(9), 848-855.
[26] Fries, R. C., & King, P. (2002). Design of Biomedical Devices and Systems (Vol. 119). CRC Press.
[27] Alexander, K., & Clarkson, P. J. (2002). A validation model for the medical devices industry. Journal of Engineering Design, 13(3), 197-204.
[28] Ward, J. R., & Clarkson, P. J. (2001). Design verification support for the medical device industry. In Proceedings of the 13th International Conference on Engineering Design (ICED'01), (pp. 211-218). Professional Engineering Publishing Ltd.
[29] Aguwa, C. C., Monplaisir, L., Sylajakumari, P. A., & Muni, R. K. (2010). Integrated fuzzy-based modular architecture for medical device design and development. Journal of Medical Devices, 4(3), 031007-1-10.
[30] Medina, L. A., Kremer, G. E. O., & Wysk, R. A. (2013). Supporting medical device development: a standard product design process model. Journal of Engineering Design, 24(2), 83-119.
[31] Haskins, C., Forsberg, K., Krueger, M., Walden, D., & Hamelin, D. (2006). Systems engineering handbook. In INCOSE.
[32] International Organization for Standardization. (2000). ISO 14971: medical devices-application of risk management to medical devices. ISO.

Copyright © 2015 by ASME

Standard Article

Concurrent Engineering: Research and Applications
2015, Vol. 23(3) 183–198
© The Author(s) 2015
Reprints and permissions:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/1063293X15580136
cer.sagepub.com

Interface definitions in literature: A reality check

Jakob Filippson Parslov[1] and Niels Henrik Mortensen[2]

Abstract

Companies that develop multi-technological products are challenged on their ability to obtain high product quality and short development lead times in today's highly competitive and globalized markets. One of the main reasons for poor product quality is due to unidentified or poorly defined product interfaces during the design phase leading to unintended product behavior. In an effort to reduce the lead time and increase quality, companies may apply a modular product architecture, thus enabling parallel development and maturing of modules. Achieving a successful integration of the modules at the end of a design phase requires, however, an understanding of how the modules disintegrate from an early stage. This implies having a fundamental understanding of what an interface is. Despite the apparent academic consensus on the importance of product interfaces during design, very little research has been done on the definition and perception of a product interface within engineering design research which is the objective of this article. A structured literature review of interface definitions found within engineering design literature has been carried out. The different definitions were tabulated against four key issues concerning the nature of an interface. These were later discussed with use of a case example in order to reason out the implications to design. The literature review revealed an inconsistency in the perceptions of an interface with regard to how it manifests itself, whether it is a design object, and the use of element types. These key issues were justified using a case example of a solenoid valve. In light of the findings from the literature review, it is argued how interfaces between modules as well as interfaces that reside in the tension field between different engineering disciplines may require great attention since they are subject to negotiation and interpretation between disciplines, which could lead to miscommunication and inefficiency.

Keywords

interface, interaction, definition, multi-technological, product development, engineering design

Introduction

Problem statement

During the past century, products have become more and more multi-technological in order for companies to achieve superior product functionality (Fotso and Rettberg, 2012). However, designing products that meet the intended quality within an accepted time span is not as straightforward as it sounds when it comes to multi-technological products. One of the main reasons why companies experience poor product quality is due to unidentified or poorly defined product interfaces being discovered too late in the projects which may lead to unintended product behavior (Grady, 1994; Kapurch, 2007). The integration of multiple technology domains in today's products raises the level of complexity, and thus contributes to the challenge of identifying and defining the interfaces.

In order to provide the reader with an idea of *what* an interface is, we briefly introduce a preliminary interface definition: *an interface defines a functional or physical relation between two mating system elements across which interaction may occur*. This definition may serve as an initial working assumption for further reading.

It is a well-known fact within academia that interfaces play an important role in executing effective

[1]New Product Development, R&D, Radiometer Medical ApS, Brønshøj, Denmark
[2]Section of Engineering Design and Product Development, Department of Mechanical Engineering, Technical University of Denmark, Lyngby, Denmark

Corresponding author:
Jakob Filippson Parslov, New Product Development, R&D, Radiometer Medical ApS, Åkandevej 21, 2700 Brønshøj, Denmark.
Email: jakob.parslov@radiometer.dk

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

product development and obtaining proper product quality. Ullman (1992) states that most design efforts occur at the connections between components, and attention to the interfaces and the flows through them are important to product development. Ulrich (1995) argues that specification of the interfaces is a key part of the product architecture and the modularization effort. Bruun et al. (2014), however, concludes that there seem to be different understandings and definitions of modularization, many of which claim the need for well-defined interface. Ericsson and Erixon (1999) support this by stating that interfaces have a vital influence on the final product and the flexibility within the assortment. Other authors such as Blyler (2004) and Sundgren (1999) introduce the term interface management as the process of defining the physical interfaces between subsystems and as a powerful tool to manage systems. Sundgren (1999) further highlights the importance of having standardized interfaces in platform-based product families. This is supported by Meyer and Lehnerd (1997) who argue that managing internal and external interfaces is as important as building new functionality into major subsystems. Kihlander and Ritzén (2012), who did a study of the conceptualization phase of a Swedish car manufacturer, underline the need for better interface definitions early in product development because subsystems too often did not interact properly together on a complete product level. Hamraz and Hisarciklilar (2013) further emphasize that the interconnectedness of parts in engineering systems tends to propagate engineering changes through the system which make up typically 20%–40% of total R&D spending. Van Wie et al. (2001) also argue that interfaces are design drivers because of the close connection between the number of interfaces and the assembly cost. Hölttä-Otto and De Weck (2007) state that well-defined modules with simple interfaces can ease project management due to decoupling of tasks and provide design freedom within a module.

Despite the consensus in academia that interfaces are important to control during product development, very few authors have dealt with the very fundamental understanding of interfaces—that is, *How are product interfaces defined in literature? How are they perceived? How does an interface manifests itself in "reality"? What is it that interfaces?* In this article, we seek to answer these questions.

What would we gain from answering the above questions? A clear *interface understanding* and *a common language* across engineering disciplines are keys to successful design synthesis where a functional decomposition of the product is translated into a physical architecture. In an effort to manage complexity and increase development efficiency, concurrent engineering may be utilized through use of a modular architecture. However, concurrent engineering may only benefit efficiency if the right quality is obtained which means that the modules must integrate properly. A successful integration phase may, however, be dependent on a proper understanding of how the modules disintegrate in the first place during synthesis. A rigorous interface definition should thus support the disintegration between modules on both a functional and a structural level and provide a common language across different technology disciplines.

Take, for example, a direct current (DC) motor that is usually considered an off-the-shelf electromechanical component. Consider, if the DC motor as a technical system was to be designed by two different technology domain specialists (e.g. mechanics and electronics), then the first step would be to divide the task of designing the system in between them. Depending on the technical discipline from which the system is viewed, the system may exhibit very different function and structure. The mechanical engineer might be concerned with fastening the motor, damping vibrations, connecting the shaft to another component, and so on. The electrical engineer might be concerned with the electrical properties and supplying the correct current. On the outside, the characteristics that are being determined within each of the technical domains seem somewhat decoupled; however, we may find areas where the two domains are actually highly dependent. A greater power output from the motor may require a bigger coil, which would take up more space and possibly inducing changes to mounting interfaces or the spatial requirements in the mechanical domain. Also, if the size and weight change, the resonance spectrum might be different, thus requiring other damping mechanisms. Or if the motor is over-constrained in an off-axis position, the motor might require more power to do the job, thus dissipating more heat into a possibly temperature regulated environment.

There may be many such examples in a complex multi-technological product, where decisions made in one technology domain affect other domains. Hence, the question of how to divide the task or disintegrate the modules is not clear and may be the source of great inefficiency. It essentially all comes down to how we define and perceive the product interface as it undergoes product development.

Purpose of this article

This article provides an overview of how the term *interface* is defined in academia. We do this by reviewing the literature for definitions of the term and subsequently classify the perceptions according to four key issues.

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

Through a case example, we will argue how these different classes of perceptions (definitions) of an interface may challenge the way by which different engineering disciplines speak and communicate about a complex multi-disciplinary system.

The structure of the article is as follows. First, a section describing the theoretical starting point and the methods applied. Second, a literature review of interface definitions is presented followed by an evaluation of the findings. Finally, a discussion concerning the relevance of the findings will be presented using a case example from a medical device company.

Theoretical basis/method

Theoretical basis

The authors of this article acknowledge the Theory of Technical Systems (TTS) by Hubka and Eder (1988) and the domain theory by Andreasen (1980) as a basis for understanding products. As such, our view on products may be somewhat colored by a mechanical mindset. The intention is, however, to adopt a more broad interpretation of a product, so that also electrical, fluid, thermal, and software can be considered as part of a technical system.

The system thinking as such is recursive, which means that the system itself can be considered as an element of another system. It also means that the system can be further decomposed into subsystems revealing new elements and relations.

The way in which a system is decomposed into subsystems may be determined by the decomposition viewpoint which is affected by a team's or a person's past experience, education, discipline, and so on (Hölttä-Otto et al., 2014). Interfaces, depending on how you interpret an interface, are created as a result of composing the subsystems into a greater whole. In order to clarify the concept of an interface, this literature review will consider all interpretations of product interfaces, no matter if they are considered as functional and/or structural, between systems, elements, functional units, modules, components, parts, and so on.

Key issues

The object of analysis in this review is the nature of interfaces as a term and concept in engineering design literature. In this respect, four key issues will be addressed:

1. Perception of the interface manifestation
2. Distinction between an interface (structural) and interaction (functional)
3. Perception of an interface as part of the elements in a system or as a design object
4. Types of elements used in the definition of an interface

An important distinction should be made between the concept of an *interface* and the activity of *interfacing*. The four key issues above relate to the nature of the interface as a concept. The justification behind them stems from an understanding of the interfacing activity. This will be further treated in the discussion of the results. This article does thus not attempt to identify the most complete or "correct" definition; however, the goal is to clarify the differences and discuss the implications to design practice supported by a case example.

Method

Objective. The objective of this article is to present the results of a structured literature review on the definitions of product interfaces.

Sources. The collected literature was extracted using the search engine Scopus due to its underlying and comprehensive database of relevant sources.

The literature search initially only focused on journals, in order to capture the highest quality data. This scope was later expanded to include conference articles as well, since the amount of journal articles which specifically contained interface definitions seemed quite scarce.

Method for finding literature. A keyword search was applied to get a rough selection of articles. The following search string of keywords was used including wildcards as indicated with an asterisk: mechatronic* OR product* AND design* AND (interface* OR Interaction*) AND system* AND modul* AND complex. The wildcards allow for inclusion of various endings over a particular stem of a word. For example, modul(e), modul(ar), and modul(arity). As subject area *Engineering* was chosen and only English articles were selected. In order to roughly narrow down the search results, 48 clearly irrelevant keywords were excluded from the search eliminating the articles from which they originate from the search result. Approximately 380 articles were thus found on the basis of this keyword search.

In order to narrow down the number of articles even further, a review of the titles was performed. The criteria for excluding articles were based on the author's opinion on the article's relevance to engineering design research. This revealed a basis of approximately 100

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

articles. After reviewing the abstracts, the number of articles was reduced to approximately 25 core articles that were reviewed intensively.

A backward search and forward search were used ad hoc as the review progressed, revealing other relevant references. This last process step was less systematic and based purely on the judgment of the author.

Method for reviewing definitions. In order to address the four key issues, certain parts of the definitions were of particular interest. First, the descriptor used to articulate the nature of the interface was noted, for example, a boundary and a point. This was used to characterize the perception of the interface manifestation. To address the second key issue, it was noted whether the authors characterized the interface as consisting of a single or two entities, for example, a boundary and a pair of mating faces. The third key issue was analyzed by looking for keywords such as *functional*, *transfer of (material, energy, information)*, *structural*, and *physical*. The fourth issue was a matter of listing the different denotations of elements.

Method for classifying definitions. The classes were initially created based on the authors' assumptions of how different technology domain specialists would think of an interface. During the reviewing process, an affinity-diagram (Beyer and Holtzblatt, 1997) was established to organize the definitions accordingly and to grow confidence in the chosen classes. Every definition was analyzed according to the context from which it was derived in order to minimize bias stemming from the interpretation done by the authors.

Review of interface definitions

One of the earliest definitions found of an *interface* dates back to year 1882 where it was denoted as

> (…) a face of separation, plane or curved, between two contiguous portions of the same substance. (Oxford University Press, 2013)

This definition later evolved around the 1960s to become much more context specific—for example, used to describe phenomena in organizations, corporate strategy, history, information technology, society infrastructure, economy, and nature (Oxford University Press, 2013). The above definition is meant as a reminder of how versatile this term really is. The context of this article is within engineering design research, which is why the definitions included in this review are specifically focused on *product related interfaces*. We will, however, refrain from using this wording throughout the rest of the article and simply call it an *interface*.

Ullman (1992) with his book on "The Mechanical Design Process" builds upon the theoretical foundation of Hubka and Eder (1988) and Pahl et al. (2007) and view products as technical systems. Ullman (1992) provides the following definition of an interface:

> the boundary area between adjacent regions that constitutes a point where independent systems of diverse groups interact.

With this interface definition, Ullman (1992) includes interaction as part of the definition of an interface, thus creating a strong relationship between the two terms.

The use of boundary area and point as denotation of an interface leads us to the question of the manifestation of an interface: *Is an interface physical or immaterial? Does it have a size or volume? If so, where does the interface start or end?*

Ullman (1992) states that "(…) functions occur in the interfaces between components" and that interfaces "are the means through which the product will be designed to meet the functional requirements."

Ullman (1992) thus considers interfaces as a facilitating mean that enables an interaction between two elements. Ulrich K. (1995) provides the following interface definition, which clearly states that interfaces are physical:

> By definition, interacting components are connected by some physical interface. Interfaces may involve geometric connections between two components, as with a gear on a shaft, or may involve non-contact interactions, as with the infrared communication link between a remote control and a television set.

Ulrich claims that no interaction occurs without the existence of a physical interface. So even though he recognizes non-contact interactions, for example, infrared (IR) connection, he claims that without a physical interface, it would not be possible, that is, the IR transmitter in the remote control and the IR receiver in the television set as representative of an interface. In this sense, Ulrich argues for interfaces as being composed of two sides, whether it is geometric connections or an IR transmitter/receiver connection.

The physical perception of an interface is derived from Ulrich's believe that the specification of interfaces between *physical components* is part of product architecture. Interaction is modeled in a functional view and thus kept separate from an interface. He uses a mechanical trailer example to support his ideas.

According to Bettig and Gershenson (2010), most research about interfaces today regard the interface as

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

part of the module or component rather than having the interface represented between facing elements as an externalized entity. These are two fundamentally different beliefs that may both be right. The choice of belief may, however, have downstream implications to information models. For example, compatibility checks in computer models may require the comparison of two entities thus calling for a split interface between two elements.

According to Miller and Elgård (1998), one must distinguish between an interface and an interaction. Although they acknowledge that these two subjects need further exploration, they come up with a definition of both an interface and an interaction:

> Interfaces are the boundaries of the modules facing each other. Some relevant types of interfaces are: 1) Functional interfaces which follow the allocation of functionality. 2) Mechanical interfaces, like connectors, plugs, surfaces, etc. 3) Electrical interfaces, like communication, signals, or power. (Miller and Elgård, 1998)
>
> Interactions describe the input/output relations between modules. Also the input/output relations between modules need to be compatible. (Miller and Elgård, 1998)

Miller and Elgård (1998) investigate the phenomenon behind modularization. They do not consider modules as limited to physical entities but rather accepts that modules can represent immaterial things such as software and knowledge. Their abstract and immaterial definition of an interface as *boundaries* also reflects this, in the sense that it can be both, functional and physical. However, with their characterization of an interface as electrical (e.g. power, signal), they seem to implicitly create an overlap between their definition of an interface and interaction. They further conclude that the relation between output and input needs to be compatible; however, the term compatibility is not further clarified.

Lam and Shankar (1994) provide some reflections on how an interface is considered within the software domain. They state that

> A physical interface occurs where a module and its environment interact. For different kinds of physical interfaces, such interactions take on a variety of forms. For a vending machine, an interaction may be the insertion of a coin. (Lam and Shankar, 1994)

In other words, an interaction between a module and its environment is a precondition for the existence of an interface. A mechanical engineer may argue otherwise—that the interface is a precondition of an interaction. This discrepancy in mindset may be a reflection of the fundamental difference between software and hardware, where software has a heavier focus on control, and therefore argues that interface follows interaction. Lam and Shankar (1994) thus also reflect on where interface information lives and how it is controlled. They view an interface as having two sides. Each interface has a service provider on one side and a service consumer on the other side. Interface interactions between a module and its environment are modeled as discrete events and each event in an interface is explicitly defined to be under the control of the service provider or consumer of the interface (Lam and Shankar, 1994).

Sellgren (1998) writes that *subsystems interact at common interfaces, where an interface is a pair of mating faces*. Hence, whereas the interaction is common, the interface can be considered as a pair of distinct faces. Blackenfelt and Sellgren (2000) further add, based on the definition by Sellgren and Andersson (1998), that *an interface may be defined as a pair of mating faces between two elements. i.e. a module interface is a pair of mating faces between two modules*. In this definition, Blackenfelt and Sellgren (2000) translate the elements into modules thus making the interface physical; however, their use of *system* does not reveal whether the interface exists between functional and physical elements. The following statement reveals their view of an interface as a physical entity:

> (…) an interface is hardly a separate entity that may be designed or optimized isolated from the components or modules. Since a mating face constitutes a part of a component, the component may be changed without changing the mating face however the mating face may not be changed without changing the component. (Blackenfelt and Sellgren, 2000)

With this statement, Blackenfelt and Sellgren (2000) thus deduce that an interface is a physical entity and not something immaterial that can be manipulated separate from the element.

In order to be able to address interfaces as a separate object, they introduce the concept of a "black box volume," which may be assigned with characteristics such as maximum size, possible location, and known dimensions early in the development phase (Blackenfelt and Sellgren, 2000). Defining interfaces as a separate object thus makes information about them more explicit. It seems as if the definition is fitted to the mechanical nature of the article, which is quite mechanical in nature.

Later we find that Blackenfelt (2001) widens the definition of an interface to include function:

> the mating faces between two modules, where mating faces have a wider meaning than physical contact. The interface may be defined in various domains where the functional

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

> relations (E, I, M) give a description at the function/solution level, whereas geometry, space and other more detailed descriptions may be used at the solution/part level.

Blackenfelt reflects upon interfaces from two different domains: function and part. This corresponds to the organ and part domains in the domain theory (Andreasen, 1980). Hence, he explicitly define *mating faces* as not just physical but also functional. The functional interface may in that sense include interacting properties.

Sellgren and Andersson (2005) introduced what they call functional interfaces, which are *the interfaces that realize or implements the different technical and interactive functions of a product*. They define a functional interface as

> (…) an intended interaction relation between two functional surfaces. (Sellgren and Andersson, 2005)

They further classify two types of functional interfaces as

> Technical interface—an intended interaction relation between a pair of technical functional surfaces in or on a technical system or in the environment. (Sellgren and Andersson, 2005)
> Interactive interface—an intended interaction relation between an ergonomic or communicative functional surface on a technical system and a sensory feature of a real or generic human. (Sellgren and Andersson, 2005)

The use of the term *functional surface* in the interface definition comes from Tjalve (1979) and is based on a mechanical conception of a system. Sellgren and Andersson's interface definition thus seems to be based solely on mechanical systems. It is also worth noting that the entities of the interface as denoted by Sellgren (1998) have now changed from being *mating faces* to *functional surfaces* both between systems.

Van Wie et al. (2001) define the interface as

> a spatial region where energy and/or material flow between components or between a component and the external environment.

Van Wie et al. (2001) argue that their definition is somewhat simplified because they only include the most "fundamental flows" such as material and energy. They argue that information is redundant with material and energy flow, since information is a subset of those two. Also, spatial and structural aspects of interfaces are simply refined physical descriptions. Similar arguments have been found in work by Dickerson and Mavris (2010), Hubka and Eder (1988), and Andreasen (1980).

Van Wie et al. (2001) talk about flow between components and represent an interface as a spatial region. This again points to the question of interfaces as something that takes up space like Blackenfelt and Sellgren (2000) argues.

While Van Wie et al. (2001) may have tried to narrow the definition a bit in order to make it more operational, Kapurch (2007) provides a somewhat broader definition of an interface:

> An interface is any boundary between one area and another. It may be cognitive, external, internal, functional, or physical. Interfaces occur within the system (internal) as well as between the system and another system (external) and may be functional or physical (e.g., mechanical, electrical) in nature.

This definition by Kapurch (2007), as documented in the NASA Systems Engineering Handbook, seems to be more inclusive and abstract in the sense that they define an interface as any boundary between one area and another. In opposition to Ullman (1992) that defines the interface as a boundary area, Kapurch (2007) simply calls it a boundary, leaving out area from the definition. In that way, they avoid commenting on the physical realization of the interface, which makes the definition somewhat more generic.

Stating that an interface may be functional in addition to physical suggests that interfaces also possess interacting properties such as the transfers of material, energy, and information. Based on this, one might argue that because interfaces possess interacting properties, it is not without importance where the interface is located, hence to what the specifications refer to, because, for example, flow properties as a measure of interaction have a tendency to change over time and space.

The reason why the definition is so broad may be due to the context in which it is used. NASA Systems Engineering Handbook attempts to capture both the product- and the process-related aspects of interfaces through the products' life cycle. Kapurch (2007) thus adopts the definition that seems fit within the purpose of the context.

The American Department of Defense (DoD) has contributed with several definitions of an interface from 2000 to 2008:

> The functional and physical characteristics required to exist at a common boundary or connection between systems or items. (United States Department of Defense, 2000)

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

> The performance, functional, and physical characteristics required to exist at a common boundary. (United States Department of Defense, 2001)
> A boundary or point common to two or more similar or dissimilar command and control systems, sub-systems, or other entities against which or at which necessary information flow takes place. (United States Department of Defense, 2008)

The first definition from 2000 considers an interface as a *common boundary*, which seems to indicate that the functional and physical characteristics are shared by both interfacing elements and are not specific to either side. The second definition from 2001 is somewhat similar except for the addition of *performance characteristics*, which seems to relate to the goodness of the solution.

The third definition from 2008 seems to be more focused on information systems and does not distinguish between required and non-required characteristics. Their perception of an interface as a common boundary is immaterial in nature and externalized from the interfacing elements. None of the documents seems to elaborate on what characteristics allow for compatibility as well as what characteristics are required and non-required.

It seems as if the DoD definition of an interface has developed from more generic to being more minded for software control systems given that they specifically regard information flow to take place at the interface. In the most recent DoD publication of Military definitions from 2014, the interface definition is omitted. Despite attempts, it has not been possible for the authors to retrieve an explanation as to why is was omitted.

The International Organization for Standardization (ISO) standard on Information Technology defines an interface as follows:

> A shared boundary between two functional units, defined by various characteristics pertaining to the functions, physical signal exchanges, and other characteristics. (ISO/IEC 2382-1:1993, 1993)

This definition also considers an interface as being a shared boundary between the two elements that are called functional units. Using the words *functional units*, they stress the fact that units within information technology are not physical as such. They do, however, acknowledge that software is communicated through physical signals. By letting the interface describe characteristics pertaining to the signal exchange, they implicitly state that interfaces characterize interactions. A shared boundary indicates that the interface is considered as an object that can be designed.

Liang and Paredis (2004) apply the concept of ports to model interfaces of a system:

> Ports are defined as locations of intended interaction between a component and its environment. Together they constitute the interface of a component, and define its boundary in a system configuration.

A system interface is defined by a composed set of ports. Each port represents locations of intended interactions. In order to connect one system with another, two ports from each of the systems must be connected. However, as it is defined here, the interface only refers to the composition of ports of a single element.

Rahmani and Thomson (2012) also provide thoughts on interfaces and the opportunities that being in control of interfaces opens. They base their article on the concept of "ports" as described above by Liang and Paredis (2004) to model places of intended interaction.

Rahmani and Thomson (2012) distinguish between interfaces and interactions in the following definition:

> An interface refers to any logical or physical relationship required to integrate the boundaries between systems or between systems and their environment. Here the word "system" refers to a set of interoperable elements compatible with each other in form, fit and function to achieve a specific outcome. Interfaces can be regarded as places where the boundaries of two subsystems come together. The places of intended interactions among subsystems are called ports. (Liang and Paredis; 2004; Rahmani and Thomson, 2012)

By defining the interface as being a relationship of either logical or physical character, Rahmani and Thomson (2012) target both systems of software and hardware. Thus, in their definition, an interface is not a set of boundaries in itself but rather a relational circumstance, whether it be logical or physical, which is required in order to tie together two system boundaries. By describing the interface as a relationship rather than a physical entity, Rahmani and Thomson (2012) impute the term with an abstract meaning. Also, when regarding the interface as a place, they avoid commenting on whether the interface has two sides to it or not. Instead, ports are used as an abstraction of each of the interfacing elements. The port methodology, however, operates with two ports in a connection—one for each side.

Rahmani and Thomson (2012) assume three types of information to be included in an interface representation: (a) the specification of port attributes, (b) the requirements on port attributes, and (c) the connectivity relationships among ports. One might ask how to know at which point the "right" attributes have been included to ensure compatibility.

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

Baldwin and Clark (2000) and their book called *Design Rules, Volume 1—The Power of Modularity* also share some insights on interfaces and their meaning. They state that

> Interfaces (are) detailed descriptions of how the different modules will interact, including how they will fit together, connect, communicate, and so forth.

Interface is here used to describe both the actual geometrical fitting and the interaction. The choice of words in the citation above seems to view an interface more as a specification rather than an actual physical entity.

Lalli et al. (1997) have written a training manual on interface control for NASA where they argue that

> An interface is that design feature of a piece of equipment that affects the design feature of another piece of equipment.

By equipment they mean a *functional area assigned to a specific source* (Lalli et al., 1997). So a design feature may only be considered an interface, if it affects another design feature outside the equipment. The boundaries between the functional areas thus become the interfaces. They argue that the interface characteristics extend beyond the interface boundary or plane where the functional areas come together. This leads to the following conclusion:

> The interface could be affected by, and therefore needs to be compatible with, areas that contribute to its function but may not physically attach." (Lalli et al., 1997)

The interesting acknowledgement here is that compatibility of interfaces needs to be obtained from both a physical and a functional point of view. It is, however, not clear from the manual what interface characteristics that extend beyond the plane.

Mikkola (2001) describes an interface as *linkages*:

> Interfaces are linkages shared among components, modules, sub-systems of a given product architecture.

The denotation of interfaces as linkages is rather metaphoric. The keyword here is that interfaces are *shared* among the elements thus representing a separate object. Mikkola (2001) provides some examples such as tolerance specification of the components, operating frequency bandwidths, and maximum heat dissipation, to name a few. Given those examples, Mikkola (2001) must consider a shared interface description to have characteristics specific to either side of the interface.

Scalice et al. (2008) consider interfaces as purely functional:

> To the author, interfaces are functional surfaces that unite two or more modules and carry out, at least, one of these functions: provide support, transmit power, locate part on assembly, provide location for other parts and transmit motion.
> an interface is an area where there is a flow of energy, material, information or, at least, a spatial interaction among two or more modules or parts.

Again, interaction seems to be considered a prerequisite of an interface. It is difficult to evaluate the completeness of the interface functions that are listed. It seems as if the author adopts the definition that fits with the presented example in the article.

Buur (1990) is concerned with mechatronic systems and does not specifically define his view on what an interface is. Instead, he contributes with a term called an interface organ:

> (…) Internally in the mechatronic system, the split between functions realized by mechanical, electronic and software means is specified by interface organs.

His theoretical point of departure is the domain theory (Andreasen, 1980) and TTS (Hubka and Eder, 1988), which seems to be the reason for articulating a cross-functional element as an organ. The idea here is that there are certain common components that basically translate between the different technology domains in mechatronic products, for example, volume controls, keyboards, and microphones. These cross-functional integrated components thus become physical interface organs because of their inherent translational functionality.

Hoffman (1990) characterizes the interface as

> (…) a module interface (hereafter just interface) as the set of assumptions that programmers using the module may make about its behavior. An interface specification is a statement, in some form, of these assumptions.

It is clear that the nature of software affects the perception of the interface. An interface thus describes the behavioral properties of a module. Translating this into a mechanical domain would mean that the interface should describe the functional interactions across the interface. The perception of an interface within software seems more one-dimensional.

Sage and Lynch (1998) describe 10 types of interfaces: internal, external, function, physical, logical, environmental, dynamic, hardware to hardware, software, and hardware-to-software interfaces. The different types of interface categories do not seem mutually exclusive and seem to be based on pragmatism. That is, an interface could both be internal and functional. Sage and Lynch (1998) argue for an interface specification

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

language that will cover the above-mentioned types, in order to describe the interface at its most fundamental level. However, they do not go further into this.

Grady (1994) provides an elaborate notion of an interface in his book on Systems Integration:

> An interface is a plane or place at which independent systems or components thereof meet and act or communicate with each other. An interface is characterized by two terminals, each touching one element in the system architecture. An interface is completed between these terminals via an interface media such as physical contact, electrical signals in wiring, fluid flow in plumbing, or a radio signal in space.

The interface is here perceived as an immaterial *plane* externalized from the elements. The interface is perceived as having two sides—a terminal for each element. Grady further argues that the interface is only completed via an interaction between the terminals.

Prasad (1997) argues for the importance of identifying the amount of interface data that are common and must be shared between work groups and provides a few examples of interfaces:

> In software design, an interface may be the definition of a procedure; in electrical design, the interface might be the external pins of a circuit. In mechanical design, the interface is usually some portion of the geometry defining the boundary of a part.

The nature of interfaces as described is very different across the engineering disciplines. Whereas the perception of an interface within the electrical and mechanical domains is very physical, the software domain is much different and more functional. According to Prasad (1997), it is possible to divide a module into an interface portion and an independent portion, which will reduce the risk of conflicts because only the interface is shared. It is not further described how to determine the interface portion from the independent portion.

Jarratt et al. (2004) developed a *linkage model* for a diesel engine where they defined eight classes of *linkages*: mechanical steady state, mechanical dynamic, spatial, thermal steady state, thermal dynamic, electrical signal, electrical earth, and electrical dynamic. They state that *Geometric linkages (mechanical steady state and spatial) are bi-directional, but the other six could be uni-directional depending upon the components involved* (Jarratt et al., 2004). By this they indicate that the elements between which the *linkages* are defined have critical impact on the information which the *linkages* can contain. They also set up certain decomposition rules, for example, *if there is a gasket between two components then half the gasket is considered to belong to each component (...)*, or *if a component is in contact with a gas or a fluid then assume that half the gas or fluid belongs to that component* (Jarratt et al., 2004). This need to carve the system in equal halves seems to be driven by physical mindset of a system.

Evaluation of interface definitions

The following section will summarize the results of the literature review according to the four key issues. In relation to the first key issue, the literature review revealed 13 different perceptions of the manifestation of an interface. They vary from very abstract perceptions such as a boundary to very concrete manifestations such as a physical geometric connection. Some of them represent clusters of perceptions that have been evaluated as having similar characteristics: a boundary, a boundary area, and a plane. Illustrations of the different perceptions have been drawn (see Figure 1(a) and (b)) in order to support the notion that the perception of the nature or manifestation of an interface is different across the definitions.

As can be seen from the sketches, most of them are metaphorical in their representation of the interface perceptions and some are very specific examples. Hence, these are not meant as exact representations but merely as supporting images for the reader of this article to reflect upon. Maybe the reader has other mental images of the same 13 categories, which is exactly the point that the concept of an interface is colored by the observers own experience, conceptual and educational background. Bucciarelli (1994) calls this aspect a matter of different engineering disciplines working in different "object worlds." They will perceive terms and diagrams differently (Jarratt et al., 2004) which may complicate communications and common understanding concerning the decomposition of a product and the splitting of tasks during product development.

Table 1 presents a further classification of the different perceptions according to the four key issues. The literature references have been plotted in the matrix to maintain traceability and to indicate how the different interface definitions are configured. A definition may be represented several times due to the composite structure of some of the definitions.

Related to the second key issue, the majority of authors believe that system relations can be viewed from both a functional and a structural viewpoint and therefore that interfaces can be both functional and structural. Capturing both functional and structural interfaces is key in an effort to disintegrate modules and enable concurrent engineering (Ulrich and Eppinger, 2012). A functional interface is often described as functional transfers of material, energy, information, and spatial relations (Pimmler and

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

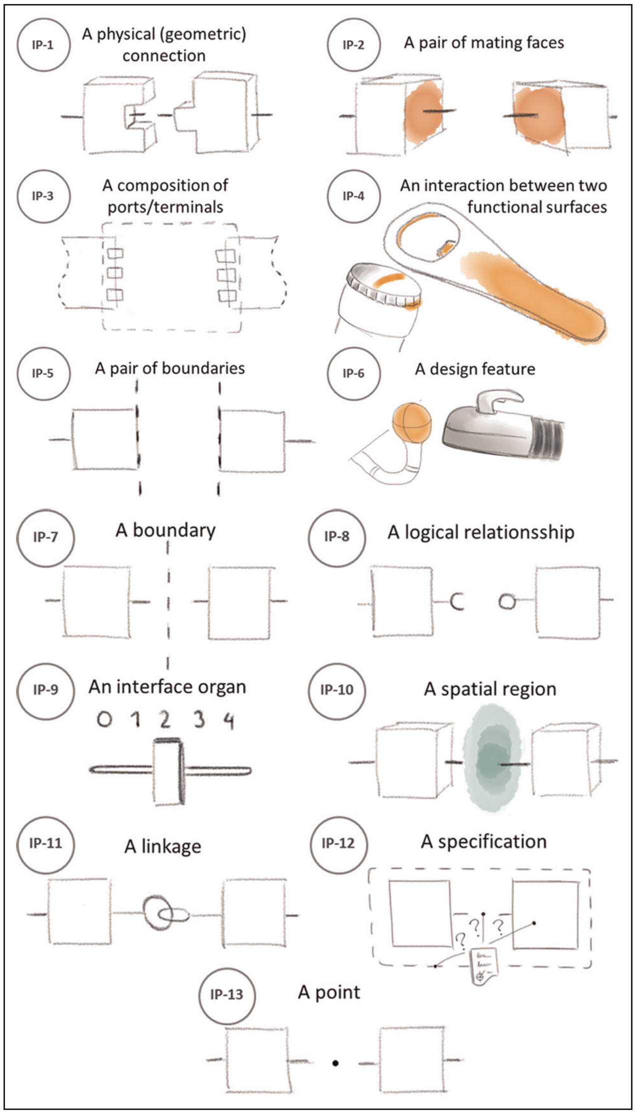


Figure 1. Illustrations of perceptions of interface manifestations as found in the literature.

Eppinger, 1994), and according to most authors these occur at the structural interface. One could argue that characterizing the interface as being both functional and structural adds to the ambiguity of the term. Based on the authors' experience, an interface is such a commonly used word in an engineering organization that people tend to underestimate the richness of meaning which the term embodies and may implicitly take for granted their own conceptual viewpoint of the context in which it is used. This could lead to misinterpretations and miscommunication.

Another interesting point has to do with the widespread use of systems language (12 different definitions) for denoting the elements that interface (key issue 3). The naming of the elements seems to fall into three overall categories of languages: systems, functional, and structural language. Using systems language may provide a frame of reference to speak about an interface; however, since almost everything in reality can be described as a system with elements and relations, it does not provide much insight into *what* it is that *interfaces. What are the specific characteristics and properties related to the system, that can help us better understand the nature of the relation—the interface?* The interface as a concept thus seems to be relative to the system model in which it is applied. Two of the definitions use functional language (Buur, 1990; ISO/IEC 2382-1:1993, 1993), whereas seven definitions apply structural language to denote the system elements (Baldwin and Clark, 2000; Blackenfelt, 2001; Hoffman, 1990; Jarratt et al., 2004; Miller and Elgård, 1998; Scalice et al., 2008; Ulrich, 1995). Six of the definitions use a mixed set of languages to denote the system elements.

With regard to the fourth key issue, the different perceptions have been classified into two classes: (a) one that views the interface as part of the elements and (b) one where the interface itself is considered an object to be designed and controlled. The difference between the two classes has to do with the level of abstraction with which you view the system. Considering the interface as part of a subsystem (child of a parent system) would be an element-view of the interface where the interface information would belong to the respective subsystem and be designed as part of the subsystem (bottom–up approach). Around half the interface perceptions view this to be the case. The other half of the perceptions supports the notion that an interface is a design object that undergoes its own development and is derived from a parent system. This would be a top–down approach to systems design. What characterizes the A-type perceptions is the fact that they seem to indicate that the interface is itself composed by two entities, each belonging to an element. B-type perceptions, however, resemble a symmetric entity that separates elements.

The relevance of this classification relates to the *interfacing* activity and the aspect of establishing compatibility at the interface when there is a difference of ownership and diverse disciplines involved. This will be further treated in the discussion.

In summary, the review has revealed a lack of consensus between the definitions with regard to three out of the four key issues, namely: (1) perception of the interface manifestation, (3) understanding of the interface as a design object and (4) the types of elements used in the definitions.

The review of the definitions have focused on the nature of an interface as a concept through the use of four key issues. However, in order to understand why these key issues matter in design practice, a discussion

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

Table 1. Overview of how different interface definitions relate certain perceptions of an interface to certain types of system elements (the perceptions consist of consolidated descriptions extracted directly from the definitions).

		Perceptions of an interface												
		Part of element						Design object						
		Physical	Dual viewset (physical or functional)											
		A physical (geometric) connection	A pair of mating faces	A composition of ports/ terminals	An interaction between two functional surfaces	A pair of boundaries	A design feature	A boundary/a boundary area/a plane	A logical relationship	An interface organ	A spatial region	A linkage	A specification	A point
	Naming of elements	IP-1	IP-2	IP-3	IP-4	IP-5	IP-6	IP-7	IP-8	IP-9	IP-10	IP-11	IP-12	IP-13
Systems language	Systems/ subsystems/ elements/ entities/items		Sellgren and Andersson (1998); Blackenfelt and Sellgren (2000)	Grady (1994)	Sellgren and Andersson (1998)	Rahmani and Thomson (2012)	Lalli, Kastner and Hartt (1997)	Kapurch (2007); United States Department of Defense (2000, 2008); Liang and Paredis (2004); Grady (1994)	Rahmani and Thomson (2012)			Mikkola (2001)		Ullman (1992)
Function language	Functional surface/ functions/ functional units/area/ functional area				Sellgren and Andersson (1998)		Lalli, Kastner and Hartt (1997)	Kapurch (2007); ISO/IEC 2382-1:1993 (1993)		Buur (1990)				
Structural language	Modules	Lam and Shankar (1994)	Blackenfelt (2001)		Scalice, Andrade and Forcellini (2008)	Miller and Elgård (1998)						Mikkola JH (2001)	Baldwin and Clark (2000); Hoffman (1990)	
	Components/ parts/body	Ulrich (1995); Prasad (1997)	Sellgren and Andersson (1998)	Liang and Paredis (2004)	Scalice, Andrade and Forcellini (2008)						Van Wie, Greer, Campbell, et al. (2001)	Jarratt, Eckert and Clarkson (2004)		
Misc	Regions/groups							Ullman (1992)						Ullman (1992)
	Environment	Lam and Shankar AU (1994)									Van Wie, Greer, Campbell, et al. (2001)			
	[No mentioning of element]							United States Department of Defense (2001)					Prasad (1997)	

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

around the activity of interfacing will be presented in the following section.

Discussion

When synthesizing a complex multi-technological system and modularizing it with a specific purpose, the module interfaces become highly important to manage since they govern the functionality of the module. A module may be considered as an aggregation of parts with a highly integrated pattern of internal functional and structural interfaces. The external module interface may also be considered as composed of several structural connection points and different functional interactions—some of which are highly related to the functionality of the system and are thus strategically important to protect and control.

To frame the discussion of the relevance of the key issues, we may therefore imagine two kinds of interfaces in a modular product: A-type—interfaces between modules, where the likelihood of variation is high due to, for example, future upgrades, serviceability, and maintainability. They have strategic importance and thus require a rigorous definition of both the functional and structural characteristics of the interface. B-type—interfaces between components such as physical contact points with transfer of work, current, and heat. These interfaces would be driven by aspects such as reliability or robustness and may be important to functionality but may be designed by only a small team and not have the same level of strategic importance as the A-type.

There may be >1000 of these B-type interface in a complex system where only 20–30 of them are promoted to A-type interfaces. This assumption will provide the frame of reference for the following discussion of the key issues.

In the following, we present an analysis of a solenoid valve system used for flow regulation in a blood gas measuring instrument (medical device). The function of the valve system is to open for or cut-off a flow of gaseous substances flowing into the instrument. The solenoid valve is a mechatronic system with physical parts actuated by an electromagnetic coil that again is controlled by a digital-to-analog electric circuit and control software to generate the digital clock signal. This mechatronic system was developed to satisfy specific requirements such as physical size, minimum heat dissipation, and timing, among others.

This rather simple system may from a structural viewpoint be considered to consist of two modules: (a) an actuator module that provides the function of generating a translational motion and (b) a valve module that has the function of sealing off or opening for a material flow. Each module thus encapsulates a certain functionality but is described by its structural composition of parts (see Figure 3).

In Figure 3, the two modules and their parts are illustrated. As one can tell from the illustration, there are quite a number of interfaces within each module and few between the modules. The interfaces are of both functional and structural nature and a few of them have been highlighted for the purpose of the following discussion (see Table 2).

As illustrated in Figure 2 and described in Table 2, this system contains a number of B-type interfaces which has to do with concentric fixating the components, guiding the anchor in a linear motion, transmitting an electromagnetic field from the coil to the anchor. However, we also see an example of an A-type interface (see "relation c" in Figure 3) between the anchor and the rubber. One could argue that this is a B-type interface that is promoted to an A-type interface because it realizes the critical interaction between the two modules allowing the system to function as intended.

In relation to the fourth key issue, this type of interface may favor a perception of an interface as a design object, where it is crucial that both the functional and structural aspects of the interface are considered. Being a design object means that it must be under design control and therefore systematically be specified, designed, verified, and validated. In addition to that, it must be documented in a Product Data Management (PDM system) in order to keep track of the maturity status, versions, revisions, and ownership. Promoting an entity to a design object is therefore not a trivial decision, which is why it would make sense only to promote some interface as design objects—hence the distinction between type A and type B interfaces.

Type A interfaces are therefore arguably more important than B interfaces due to their role of realizing module interactions. But another important point has to do with the multi-disciplinary aspect of interfaces. As we indicated in the evaluation of the definitions, when engineers from different disciplines reason about products they are biased by their respective design experiences, educational backgrounds, and conceptual viewpoints. They have different mental models of the entities they are concerned with (Jarratt et al., 2004). Interfaces are thus not exempted from this kind of interpretation. A mechanical engineer tends to have a highly visual mental model of components and their physical contact points, whereas an electronics engineer tends to think more in, for example, budgeting the input and output flows. Even within the same disciplines, there may be a myriad of different perceptions (Jarratt et al., 2004). Every discipline therefore ascribes their own meaning to the interface term that suits their own work practice and way of reasoning, thus not favoring multi-disciplinary cooperation. In some

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

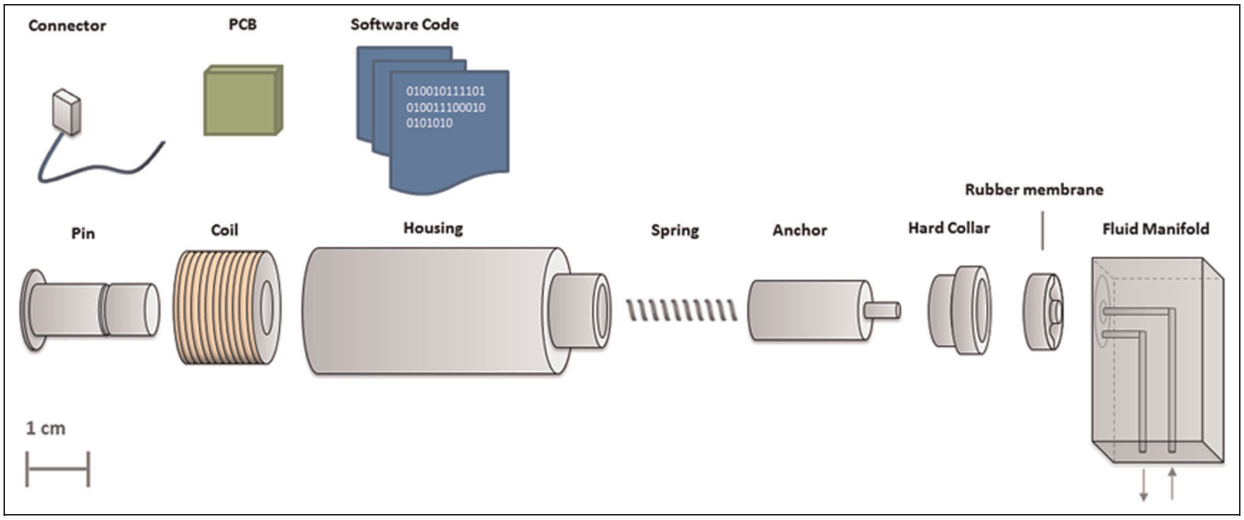


Figure 2. Assembly overview of a solenoid valve. The PCB and the software code are physically dislocated from the rest.

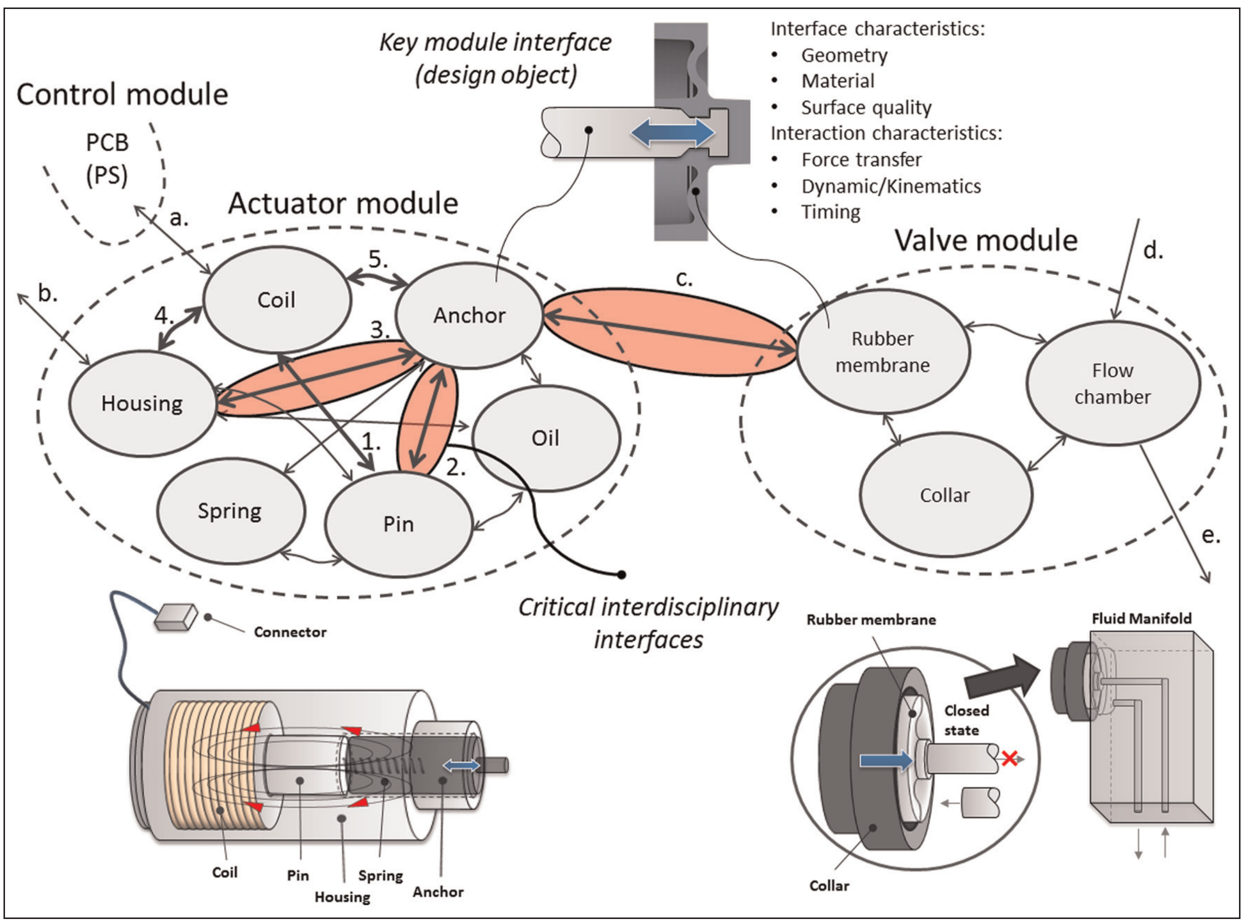


Figure 3. An analysis of the two modules and the types of interactions. The key module interface between the two modules is between the anchor and the rubber membrane. This interface could be considered a design object given its inter-modular nature.

instances, interfaces between modules or components coincide with disciplinary boundaries which mean that the interfaces are made subject to negotiation and interpretation across not only different module owners but

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

Table 2. A list of highlighted interfaces.

	Functional interfaces	Purpose of the interface
Internal	1. EM field forces	Transmit EM force field, fix position
	2. Mech. forces + EM field forces	Magnetize anchor through EM force field
	3. Mech. sliding motion (forces) + EM field forces	Guide mechanical motion
	4. EM field forces	Transmit EM force field
	5. EM field forces	Attracting anchor
External	a. Electrical power	Provide electric energy
	b. Thermal energy + mech. forces	Heat transfer due to friction, fixate housing, and collar in flow chamber
	c. Mech. forces to valve organ	Transmit motion

EM: electromagnetic.
Some of the arrows in Figure 3 capture more than one interface. For example, the second is both a mechanical and electromagnetic functional interface.

also across different technology disciplines. The question of perception of an interface (key issues 1, 2, and 3) therefore becomes relevant.

To give an example, we consider the interface between the anchor and the housing of the valve from a mechanical, electrical, and control engineering perspective. A mechanical engineer might be concerned with achieving a frictionless and accurately guided linear movement of the anchor in the housing. To achieve this, he or she would mentally produce property models that show how geometry (tolerances), surface quality, and material properties will affect the wanted frictionless properties. However, because of this structural perception of the interface, it may be that the mechanical engineer misses the impact that a change to, for example, tolerances have on the electromagnetic properties, which is the concern of the electrical engineer. For example, the air gap between the anchor and the housing (mechanical tolerances) has an exponentially large impact on the electromagnetic circuit (electrical properties). The electrical engineer on the other hand might be concerned with achieving as low a remanence level in the anchor after the power is turned off. This property, however, has relations to a number of mechanical aspects such as material properties of the anchor, air gap, surface roughness, and manufacturing process, which may not be part of the mental model of the electrical engineer. The interface thus becomes subject to interpretation between the disciplines which may lead to miscommunication and compatibility issues. This example stresses the need for a clear definition of an interface that can serve as a common language across different engineering disciplines, thus allowing for more efficient concurrent development of complex systems.

The above discussion points to the fact that some interfaces are more important to manage and control than others. Given the difference in perceptions of interfaces as found in this review, we may suggest that interfaces that are inter-modular and lie in the tension field between different engineering disciplines seem to be of critical importance to manage since they can be considered as objects between different areas of ownership and different conceptual viewpoints. This could be a subject for further investigation through empirical studies of practice.

Conclusion

This article investigates how product interfaces are defined and perceived, through a systematic literature review on interface definitions. The definitions were tabulated against four key issues in order to highlight the discrepancies between them. The review revealed 13 different perceptions of the manifestation of an interface. In addition, it was found that the majority of the authors consider an interface to be either functional or structural and around half of the perceptions were considered to belong to the elements versus being a separate design object. Another key point is the fact that there is a mixed use of different languages (i.e. systems, functional, structural languages) to denote the elements that interface. In general, there seems to be a lack of consensus concerning the nature of an interface in engineering design.

Through the use of a case example of a solenoid valve, it has been justified that the discussion of whether or not to consider an interface as a design object is relevant since it is unfeasible that all interfaces between thousands of components in a complex system may be controlled to the same degree. Thus, there may be a selection of interfaces that deserves greater attention than other. The discussion suggests two instances where an interface may need greater attention; interfaces between modules as well as interfaces that reside in the tension field between different engineering disciplines where the risk

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

miscommunication is high due to the lack of common consensus concerning the nature of interfaces.

Further research could be to perform an empirical study of the perceptions of interfaces with practitioners and to investigate how these perceptions may influence collaboration in design practice.

Acknowledgements

The authors would like to express their gratitude to Radiometer Medical ApS, for providing the case example as well as giving valuable insight into their product development activities, and Professor Emeritus Mogens Myrup Andreasen, for inspiring discussions on the subject matter.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research is part of J.F.P.'s doctoral studies which is funded jointly by the Ministry of Higher Education and Science under the Industrial PhD Programme and by Radiometer Medical ApS.

References

Andreasen MM (1980) *Machine design methods based on a systemic approach*. PhD Thesis, Lund University, Lund.

Baldwin C and Clark K (2000) *Design Rules: The Power of Modularity*. Cambridge, MA: MIT Press.

Bettig B and Gershenson J (2010) The representation of module interfaces. *International Journal of Product Development* 10(4): 291–317.

Beyer H and Holtzblatt K (1997) *Contextual Design: Defining Customer-Centered Systems*. Amsterdam: Elsevier.

Blackenfelt M (2001) *Managing complexity by product modularisation*. PhD Thesis, Department of Machine Design, Royal Institute of Technology, Stockholm.

Blackenfelt M and Sellgren U (2000) Design of robust interfaces in modular products. In: *Proceedings of ASME design engineering technical conference*, Baltimore, MD, 10–13 September, paper no. DETC00/DAC-14486. New York: ASME.

Blyler J (2004) Interface management—managing complexity at the system interface. *IEEE Instrumentation & Measurement Magazine* 7: 32–37.

Bruun HPL, Mortensen NH and Harlou U (2014) Interface diagram: design tool for supporting the development of modularity in complex product systems. *Concurrent Engineering: Research and Applications* 22(1): 62–76.

Bucciarelli L (1994) *Designing Engineers*. Cambridge, MA: MIT Press.

Buur J (1990) *A Theoretical Approach to Mechatronics Design*. Lyngby: Institute for Engineering Design, Technical University of Denmark.

Dickerson C and Mavris D (2010) *Architecture and Principles of Systems Engineering*. Boca Raton, FL: CRC Press.

Ericsson A and Erixon G (1999) *Controlling Design Variants: Modular Product Platforms*. New York: SME Press.

Fotso AB and Rettberg A (2012) State of the art for mechatronic design concepts. In: *Proceedings of 2012 IEEE/ASME 8th IEEE/ASME international conference on mechatronics and embedded systems and applications*, Suzhou, China, 8–10 July, pp. 232–240. New York: IEEE/ASME.

Grady JO (1994) *System Integration*. Boca Raton, FL: CRC press.

Hamraz B and Hisarciklilar O (2013) Change prediction using interface data. *Concurrent Engineering: Research and Applications* 21(2): 141–154.

Hoffman D (1990) On criteria for module interfaces. *IEEE Transactions on Software Engineering* 16(5): 537–542.

Hölttä-Otto K and De Weck O (2007) Degree of modularity in engineering systems and products with technical and business constraints. *Concurrent Engineering: Research and Applications* 15(2): 113–126.

Hölttä-Otto K, Chiriac N, Lysy D, et al. (2014) Architectural decomposition: the role of granularity and decomposition viewpoint. In: Simpson TW, Jiao J, Siddique Z, et al (eds) *Advances in Product Family and Product Platform Design*. Berlin: Springer, pp. 221–243.

Hubka V and Eder WE (1988) *Theory of Technical Systems: A Total Concept Theory for Engineering Design*, vol. 291. Berlin; New York: Springer-Verlag, p. 1.

ISO/IEC 2382-1:1993 (1993) Information technology—vocabulary—part 1: Fundamental terms.

Jarratt TA, Eckert CM and Clarkson PJ (2004) Development of a product model to support engineering change management. In: *Proceedings fifth international symposium on tools and methods; tools and methods of competitive engineering TCME*, Lausanne, 13–17 April.

Jarratt TA, Eckert CM, Clarkson PJ, et al. (2004) Providing an overview during the design of complex products: the development of a product linkage modelling method. In: Gero JS (ed.) *Design Computing and Cognition*. Berlin: Springer, pp. 239–258.

Kapurch SJ (2007) *NASA Systems Engineering Handbook* (ed Kapurch SJ). Washington, DC: NASA Special Publications, p. 360.

Kihlander I and Ritzén S (2012) Compatibility before completeness—identifying intrinsic conflicts in concept decision making for technical systems. *Technovation* 32(2): 79–89.

Lalli V, Kastner R and Hartt H (1997) *Training Manual for Elements of Interface Definition and Control*. National Aeronautics and Space Administration, Office of Management, Scientific and Technical Information Program. Available at: http://www.hq.nasa.gov/office/codeq/rp1370.pdf

Lam SS and Shankar AU (1994) A theory of interfaces and modules I—composition theorem. *IEEE Transactions on Software Engineering* 20(1): 55–71.

Liang V-C and Paredis CJJ (2004) A port ontology for conceptual design of systems. *Journal of Computing and Information Science in Engineering* 4(3): 206–217.

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

Meyer MH and Lehnerd AP (1997) *The Power of Product Platforms*. New York: Simon & Schuster.

Mikkola JH (2001) Modularity and interface management of product architectures. In: *Portland international conference on management of engineering and technology PICMET'01*, Portland, OR, 29 July–2 August, pp. 599–609. New York: IEEE.

Miller TD and Elgård P (1998) Design for Integration in Manufacturing. In: *Proceedings of the 13th IPS Research Seminar*. Aalborg University.

Oxford University Press (2013) Interface, n. Available at: http://www.oed.com/view/Entry/97747?rskey=Eiiy9I&result=1&isAdvanced=false#eid

Pahl G, Wallace K and Blessing L (2007) *Engineering Design: A Systematic Approach*. Berlin: Springer.

Pimmler TU and Eppinger SD (1994) Integration analysis of product decompositions. In: *ASME design theory and methodology conference*, Minneapolis, MN, September. Cambridge, MA: Alfred P.Sloan School of Management, Massachusetts Institute of Technology.

Prasad B (1997) *Concurrent Engineering Fundamentals: Integrated Product Development,* vol. II. Upper Saddle River, NJ: Prentice Hall—PTR.

Rahmani K and Thomson V (2012) Ontology based interface design and control methodology for collaborative product development. *Computer-Aided Design* 44(5): 432–444.

Sage AP and Lynch CL (1998) Systems integration and architecting: An overview of principles, practices, and perspectives. *Systems Engineering* 1(3): 176–227.

Scalice RK, Andrade LFS and Forcellini FA (2008) A design methodology for module interfaces. In: Curran R, Chao SY and Trappey A (eds) *Collaborative Product and Service Life Cycle Management for a Sustainable World*. Berlin: Springer, pp. 297–304.

Sellgren U (17–19 August 1998) Attachment of mating faces—a interrelational feature approach. In: *Proceedings of 8th international ANSYS conference*, Pittsburgh, PA.

Sellgren U and Andersson K (1998) MOSAIC—a framework and a methodology for behavior modeling of complex systems. In: *Proceeding of Produktmodeller'98*, pp. 119–137.

Sellgren U and Andersson S (2005) The concept of functional surfaces as carriers of interactive properties. In: *International conference on engineering design ICED*, Melbourne, VIC, Australia, 18 August.

Sundgren N (1999) Introducing interface management in new product family development. *Journal of Product Innovation Management* 16(1): 40–51.

Tjalve E (1979) *A Short Course in Industrial Design*. London: Newnes-Butterworths.

Ullman D (1992) *The Mechanical Design Process*. New York: McGraw-Hill.

Ulrich K (1995) The role of product architecture in the manufacturing firm. *Research Policy* 24(3): 419–440.

Ulrich KT and Eppinger SD (2012) *Product Design and Development*. New York: McGraw-Hill Higher Education.

United States Department of Defense (2000) *Defense standardization program (DSP)—policies and procedures*. No. 4120.24-M, March, Washington, DC: Defense Pentagon.

United States Department of Defense (2001) *61A (SE): Military Handbook Configuration Management Guidance. United States Department of Defense*.

United States Department of Defense (2008) *JP1–02: Department of Defense Dictionary of Military and Associated Terms*. Washington, DC: *United States Department of Defense*; Darby, PA: DIANE Publishing.

Van Wie MJ, Greer JL, Campbell MI, et al. (2001) Interfaces and product architecture. In: *ASME 2001 international design engineering technical conferences and computers and information in engineering conference*, Pittsburgh, PA, 9–12 September, pp. 1–12. New York: ASME.

Author biographies

Jakob Filippson Parslov holds an MSc in Design and Innovation from the Technical University of Denmark and is an Industrial PhD student at Radiometer Medical Aps. The project is done in collaboration with Section for Engineering Design and Product Development, Department of Mechanical Engineering at the Technical University of Denmark. His main research focus is on supporting the development of product families through an explicit focus on defining and controlling interfaces across engineering disciplines during design synthesis.

Niels Henrik Mortensen holds a PhD and an MSc in Mechanical Engineering and is employed as a Professor at the Technical University of Denmark. He is the head of section at DTU Mechanical Engineering at the Technical University of Denmark. His main research focus is procedures and methods supporting development of Product Families based on Architectures and Platforms. Currently, there are nine researchers within the field of architecture-based product development.

Downloaded from cer.sagepub.com at DTU Library - Tech. inf. Center of Denmark on February 3, 2016

Understanding Interactions in Complex Multi-Technological Products – A First Principle, Physics-based Theoretical Framework

Jakob Filippson Parslov[1], Barney Gerrard[1], and Niels Henrik Mortensen[2]

[1]New Product Development, R&D
Radiometer Medical ApS
Åkandevej 21
2700 Brønshøj, Denmark
Email: jakobparslov@gmail.com
& barney.gerrard@gerrard.dk
Tel.: +45 60152745
Corresponding author: Jakob F. Parslov

[2]Section for Engineering Design and Product Development
Department of Mechanical Engineering
Technical University of Denmark
Produktionstorvet, Building 426
2800 Kgs. Lyngby, Denmark
Email: nhmo@mek.dtu.dk

ACKNOWLEDGEMENT

We would like to express our gratitude to Radiometer Medical ApS for co-financing this research together with the Innovation Fund Denmark as part of J.F.P's doctoral studies.

ABSTRACT

Research suggests that the majority of problems during product development occur at the interfaces in a system. One of the reasons might be that products are becoming increasingly complex in terms of increasing performance and functionality by advancing technology. This leads to the involvement of various technical disciplines in the design activity and thus challenges the understanding and communication concerning interfaces and interactions. This paper presents a first-principles, physics based framework for reasoning about interactions in engineering design, which will equip system architects with a common interdisciplinary language and support them in creating unambiguous descriptions of interfaces and interactions. The initial evaluation of the framework has indicated a positive effect in terms of increasing the number of identified interactions during decomposition equalizing the difference of technical disciplines and years of experience. This framework, and way of reasoning, may significantly improve the consistency and accuracy of interface and interactions definitions in the industry.

1 INTRODUCTION

1.1 PROBLEM STATEMENT

It is commonly known that many problems during product development occur or are detected at the interfaces in a system (Grady 1994; Kapurch 2007; Wheatcraft 2010; Buede 2012). Whenever a system is decomposed into subsystems, interfaces are created and failing to identify or completely define these interfaces may be a major cause of project overruns and product failures (Wheatcraft 2010).

Complex products have many functions which are quite often realized by many different technologies and subsystems in a highly interconnected pattern. In practice that means that engineers from different disciplines have to work together to overcome this complex task of reasoning from a product's intended properties to its structural characteristics of parts and physical effects that realize these properties. Because engineers are typically taught within their own engineering domain and not taught in fundamental correlations between technical sciences, we see a lack of a common mental model and language for understanding concepts like interfaces and interactions across different disciplines, e.g. a recent literature study revealed that there is a lack of consensus concerning the perception of the manifestation of an interface. There is no common language across disciplines regarding interfaces and interaction. (Parslov and Mortensen 2015)

As a consequence, it is likely that practitioners start to use abstract language in order to communicate across disciplines about what an interface is and how it should be specified. With abstract language however, you leave space for interpretation to occur, thus introducing the risk of misinterpretation and ultimately rework. Furthermore, in many engineering companies, engineers are organized according to different disciplines and then allocated to different subsystems of the system under development, e.g. modules. In some instances, the structural boundaries (i.e. between module owners) and the interdisciplinary boundaries align in the same interface. The interface therefore becomes subject to negotiation not only between different owners but also between engineers with different technical backgrounds. Again, because there is a lack of common language, companies are challenged with how to communicate what an interface is and how to work with it.

In light of the above there is a need for a theoretical framework that allows for a useful and unambiguous communication about interfaces and interactions across different engineering disciplines both for purposes of analysis and synthesis of products. In order to scope the contribution presented in this paper, we have decided to look at the early architectural phases of product development, meaning that the focus will be on functional interactions.

The paper is structured in the following way; first related research will be presented followed by the theoretical basis and research approach. Then the *Interaction Framework* will be presented using deductive reasoning and simple examples and principle models to ease the understanding followed by a test documenting the effects of the framework. Finally, a discussion comparing the framework to other contributions in literature and a conclusion will be presented.

1.2 RELATED WORK

An *interface* as a term or concept in the engineering domain is a theoretical construct which allows engineers to speak about inter- and intra-relations between elements of a system during its development – element being of either functional or physical character. An interface is thus not an observable physical phenomenon as such and has therefore no explanatory definition of what it is. It rather boils down to how *useful* the concept is in the *context* it is being used and to the *people* that uses it. Because products are becoming increasingly multi-technological[1] and complex, the context in which the concept interface is used changes and the people using it are becoming increasingly diverse. Also, the increasing interest in Model Based System Engineering (MBSE) from both academia and industry calls for a rigorous, multi-disciplinary[2] language concerning interfaces. This is something which must be addressed in order to ensure useful employment of the term which will ultimately lead to more effective product development.

In a literature review by Parslov and Mortensen (2015) it was concluded that there seem to be a lack of consensus across engineering disciplines concerning the definition and perception of an interface. This fact was also later confirmed by Zheng et al. (2016). Specifically most authors consider an interface to be both physical and functional. Some authors view the interface to be an object in itself or part of the elements on each side. Also, to speak about relations, like an interface, it is necessary to understand the nature of the elements that the relation acts between. Thus, the elements, inform us about the type of relation. Parslov and Mortensen (2015) show that the concept of an interface is being defined using various names for the elements that *interface*; systems, subsystems, elements, entities, units,

[1] 'Multi-technological' is a characteristic of *a product* and refers to the fact that the constituent elements of the product, e.g. modules or components, are developed by multiple engineering disciplines, e.g. mechanical, electrical, software engineering.
[2] 'Multi-disciplinary' is a characteristic of *a design activity* and applies whenever more than one engineering discipline is involved in the development of a product.

modules, components, parts etc. This makes it somewhat unclear from what perspective to reason about interfaces – functional or physical – and when.

Hirtz et al. (2002) consolidated several research efforts into a functional basis for engineering design containing detailed classification of both function and flow. This work builds on earlier work by Pahl et al. (2007) who suggested three classes of functional flows; Energy, Material, Signal. They further loosely sub classified these as; 1) mechanical, thermal, electrical, chemical energy etc. 2) Gaseous, solid, fluid, human MATERIALs etc. and 3) Magnitude, display, control impulse, data, information signals. In theory of technical systems, Hubka and Eder (1988) denoted functional relations as effects which could be classified as biological objects (incl. human), MATERIAL, energy, and information. They argue that it is based on insight into various physical phenomena that allow a design engineer to synthesize a technical system by arranging these effects in a way that transforms and operand from an undesired state into a desired state.
Pimmler and Eppinger (1994) use four classes for functional relations namely; MATERIAL, energy, information, spatial. Spatial is added in recognition of the fact that the location and orientation of two system elements is important.

Liedholm (1999) also provided a classification of interactions into a second level decomposition; MATERIAL, energy nature, containing energy fields and energy flows, and information being of energy or MATERIAL nature. In the paper by Stone and Wood (2000) they introduced a design language called a Functional basis which suggested product functions to be denoted by verb-noun pairs. They also further classified both function and flow into detailed lists. Hirtz et al. (2002) later consolidated this work with work done within NIST[3] Design Repository Project (Szykman et al. 1999).

An industry domain where systems engineering and interface management have been practiced for decades is within space engineering. Lalli et al. (1997) published a training manual for elements of interface definition and control. The publication features both a technical classification of interfaces as well as a management perspective on interfaces. The four classes of interfaces are; Electrical/Functional, Mechanical/Physical, Software, Supplied services, which covered electrical power, communications, fluid, environmental characteristics. Other publications have also focused on the management of interfaces as part of systems engineering (Blyler 2004; Kapurch 2007; ECSS 2015).
More recently Bettig and Gershenson (2010) published a paper on how to represent module interfaces. They propose four classes of interfaces which are denoted by their purpose: Attachment, Control and Power, Transfer, and Field interface. They argue that these four classes involve the least duplication of effort when defining interfaces.

Other recent work has been concentrated on making information models of interfaces for use in model based systems engineering. Malmqvist (1993) addresses Bond Graphs as a way of qualitatively and quantitatively model and simulate technical system. He points out that Bond Graphs are limited to describing energy, and neglect conservation of mass, momentum and the 2nd law of thermodynamics (Malmqvist 1993). Krause et al. have developed the Module Interface Graph (MIG) for modeling interfaces in complex systems (Blees and Krause 2008; Krause et al. 2013). They argue that a new classification of interfaces needs to be developed for each type of product because of their specialized nature (Blees and Krause 2008). Liang and Paredis (2004) propose a port ontology for conceptual design of systems. In this they classified port attributes based on three system views; form, function, and behavior. *Form* represents geometric characteristics, *function* represents the intended use, and *behavior* was described using effort and flow conjugate variables. The notion of ports is intended to support systems design in that it allows you to gradually decide on form elements, once the functional attributes have been decided on. Based on this work Rahmani and Thomson (2012) published a paper suggesting a rule-based system to allow for externalization of interface logic and compatibility, for use after the conceptual design stage where design changes are many and need to be controlled.

Ulrich and Eppinger (2012) suggested a classification of interactions based on a more practical consideration; fundamental (purposeful) and incidental interactions. They argue that it occasionally is possible to reduce interactions to a well-defined interface for two chunks to implement. Further they argue that it is relatively straightforward to specify

[3] NIST – National Institute of Standards and Technology

interfaces to handle *fundamental* interactions, however it may be more difficult for *incidental* interactions, because the knowledge of the system only gradually improves over time. Grady (1994) suggested three interface types; Outerface, Innerface, Crossface which classifies the importance of a certain interface based on organizational reasons of ownership and responsibility.

Wheatcraft (2010) proposed a three step approach to identifying, defining, and writing requirements for interfaces. It is argued that anytime there is an interaction between two system elements, there is an interface and that an interface is defined as a common functional or physical boundary where two systems interact. Crawley et al. (2015) suggest viewing an interface conceptually as a system element which is described by a process (transformation function), an operand which is affected by the process and two compatible instruments, which are necessary for the process to take place. Uddin (2015) presents an Interface Analysis Framework for systems analysis which intends to support interface definitions. It consists of a template or checklist of various aspects related to interactions exchanged at the interface. The point of departure for interaction is Material, Energy, Information, Spatial, and Physical. The two latter relates to the formal aspects of the system. The Contact and Channel Approach ($C\&C^2$-A) was developed by Albers and Wintergerst (2014) which aims at associating a product's functions to its physical structure by relating functional interactions to concrete interfaces, called Working Surface Pairs. Interactions are here material, energy, information.

Zheng et al. (2016) argues for a better classification of interfaces in multi-disciplinary product development and proposes a new classification consisting of four classes; Geometric interface, Energy interface, Control interface, Data interface. However the paper does not reveal how the new classification was conceived which would have provided credibility to the classification in terms of understanding the applicability.

The above review shows that it is not clear what an interface is, how it distinguishes itself from an interaction, and whether the interface/interaction classes that are provided by numerous authors are mutually exclusive. It is furthermore not clear based on the literature how to transition from a high level notion of interaction (i.e. material, energy, information) to a more concrete level which can actually be tested and designed for.

Supporting industry in reducing ambiguity in their daily work practices around interactions must rely on an understanding of the phenomena behind the practice and the nature of the concept in question.

Based on the above review of related work it is the claim, that there is a need for a better characterization of *what interaction is* and *how a better understanding will lead to more complete interface descriptions*. More specifically we aim to answer the following research questions:

How can interactions be classified using a physics-based first principles approach, to support a system architect in reducing ambiguity during architectural decomposition of multi-technological complex systems?

1.3 THEORETICAL BASIS

The foundation for this paper lies within two different scientific fields – physics and engineering design. Both the term *interface* and *interaction* are terms found within the engineering design research domain and as stated earlier they are not very clearly separated and defined. From a physics perspective however, *interaction* is a very well-defined concept. But in order to make interaction useful in an engineering context where companies work at various levels of abstraction, and in different silos of technical disciplines, we need a new framework of understanding.

The intention of this paper is therefore to contribute to the engineering design research community by defining the concept of *interaction* based on an understanding from physics coupled with insight into the phenomena concerning the activity of *interacting*. This will allow us to devote the term *interface* to a specific meaning separate from interaction, which we will touch briefly upon towards the end of the framework.

Many of the concepts that are found in engineering design research literature are built on an understanding of the phenomena inherent in design practice and therefore driven, to a certain extent, by convenience or practicalities, e.g. most complex systems are too complicated for humans to comprehend so a well-known strategy is to divide the system into more manageable chunks and abstracting from details - how you divide and abstract the system is driven to a large extent by subjective judgement.

While we do respect and account for the phenomena of engineering design practice, which represents the context in which this framework is going to be used, we also believe that there is a need for a more objective take on the concept of an interaction and an interface, to aid the communication between different engineering disciplines. By basing the definitions in this paper on an understanding of fundamental physics coupled with an understanding of the phenomena inherent in engineering design we aim to add credibility to the *Interaction Framework* and widen the application to a broader set of engineering disciplines.

1.3.1 Physics

In order to arrive at a framework, which is applicable across most engineering disciplines, the idea has been to take a *first principles approach* and look into interaction at its most fundamental level in physics – fundamental interactions. Doing this exercise have made it possible to look beyond the bounds of various engineering disciplines and to identify the analogies and correlations that exist between them ultimately arriving at a unifying language across multiple engineering disciplines.

Throughout this paper we will touch upon a number of well-known concepts from physics such as force, momentum, energy, matter, etc. as part of the treatment of interaction. We refer to physics books (Chabay and Sherwood 2011) for a more thorough walkthrough of these concepts.

1.3.2 Engineering Design

A conscious decision has been made not to call out a certain engineering design ‘school’ as a reference point for this paper. Some examples of ‘schools’ within this research field are:

- ‘American school’ (systems engineering) (Crawley et al. 2004; Haskins et al. 2006; Kapurch 2007; Weck et al. 2011; Ulrich and Eppinger 2012; Crawley et al. 2015)
- ‘German school’ (Rodenacker 1971; Pahl and Beitz 1988)
- ‘Copenhagen school’ (Andreasen 1980; Hubka and Eder 1988; Andreasen et al. 2014; Andreasen et al. 2015)

Instead we define the necessary terms and concepts as they are needed during the paper, which are not in conflict with any of the existing theoretical engineering design frameworks. The logic behind this approach is to allow for the framework to be easily adopted by the different ‘schools’ and thus to have as broad an impact as possible in the engineering design research community.

According to Andreasen (2011) the most central behavioral characteristic of a design theory is for the theory to lead to *productive designing* through the created mindset of the designer and the models, methods, and tools. In natural sciences like physics the goal is to create better predictions of natural phenomena through modeling.

The *Interaction Framework* which is presented in this paper does not contribute to physics in terms of providing better predictions, but rather present the fundamental concepts of physics in a way that supports designers or system architects in reasoning rigorously about the possible interactions that might occur in a system independent of engineering disciplines, i.e. it contributes to design theory by leading to more productive designing.

1.4 RESEARCH APPROACH

This paper represents the result of a 3 year research effort into the nature of interactions and interfaces in engineering design and physics.

1.4.1 Research contribution

This paper has several main contributions which constitutes the framework:

1. a set of definitions; *INTERACTION*, *INTERACTION MECHANISM*, *INTERFACE*

2. a classification of *INTERACTIONS* and the *INTERACTION MECHANISMS* that facilitate them

All contributions fall within the area of engineering design research aiming at qualifying the nature of interactions and interfaces in technical systems during product development. The application is two-fold.

From an industrial perspective the framework is intended to support engineering designers with a mental model for how to reason about interactions that span across various engineering disciplines. It is therefore particularly suited for companies developing multi-technological products by means of several engineering teams from different engineering backgrounds. The specific end users are system architects who are responsible for decomposing the system and laying out the architecture top-down.

From an engineering design theory perspective the framework proposes a language for speaking and reasoning about interactions across multiple engineering disciplines. The tool is meant as a vehicle for the framework in order to operationalize the theory. Because of the rigorous approach to deducing the framework, the proposed classifications of INTERACTION MECHANISMS are both *mutually exclusive* (no overlap) and *collectively exhaustive* (no gaps). This ensures a sound and broad foundation for further research in this area.

An important note is that the framework does not aid the system architect in making the ‘right’ or purposeful decomposition of a system, but rather supports the system architect in identifying and classifying interactions and setting the requirements that these interactions place on the interfaces for each decomposition step.

Also, this framework does not attempt to discuss the relationship between interaction inputs and outputs to a system, hence the functionality of a system. Instead this framework specifically addresses how to conceptually understand and define a particular input or output of a system.

1.4.2 Research method for developing framework

The core method applied in this research has been a first principles approach to deducing the framework based on fundamental physics. The reason for applying this approach is motivated by our wish to introduce objective prescriptive support in the engineering design research community and practice which rely on a rigorous line of reasoning and argumentation. Because the line of reasoning originates from the most fundamental physics, the definitions and classification that derived from here are compatible with almost all engineering disciplines involved in product development. Hence, the framework pays careful attention to terms and definitions in order to make it easier for other researchers to adopt it in their work.

The following requirements for the framework were set up prior to the development of the framework, see table 1.

Table 1 shows the requirements for the framework, the factors that has an influence on these, the criteria which was measured for, and comments

ID#	Requirement	Influencing factor	Measurable criteria	Comments
1	The **framework** shall enable the user to **identify a greater number of interactions outside his/her area of technical expertise** than he/she would otherwise have identified w/o the framework.	User’s technical background	Number of identified interactions outside area of expertise	Support for multi-disciplinary development
2	The **framework** shall allow an **inexperienced engineer** to **identify as many or more interactions** than an experienced engineer when	User’s experience (years)	Number of identified interactions	Support for inexperienced and experienced engineers

	analyzing an existing product.			
3	The **framework** shall enable the user to **identify more interactions** than was achieved w/o the framework	User's experience	Number of identified interactions	Support for completeness of interface requirements.

It is the authors understanding that these requirements are all instrumental to achieving less integration issues, less rework, and therefore shorter development lead time and faster time-to-market. While it was not possible to test the success of the framework in a longitudinal study in industry, we have tested the four requirements in several arranged test cases. This will be explained in the evaluation part of this paper.

2 INTERACTION FRAMEWORK

In order to arrive at a classification of interactions, which is both *mutually exclusive* and *collectively exhaustive* (i.e. captures all technical disciplines relevant to engineering design), the point of departure will be fundamental interaction from physics. By starting at this low abstraction level, the reader will gain an understanding of various physical phenomena based on a simple mental ball-and-spring model (Chabay and Sherwood 2011). Reasoning at this level of abstraction however, leaves the reader very disconnected from an everyday engineering design context. We therefore describe how we abstract from the fundamentals and arrive at a classification of *INTERACTIONS* and *INTERACTION MECHANISMS*, which is true to the fundamental physics but more *useful* to engineering design. See fig. 1.

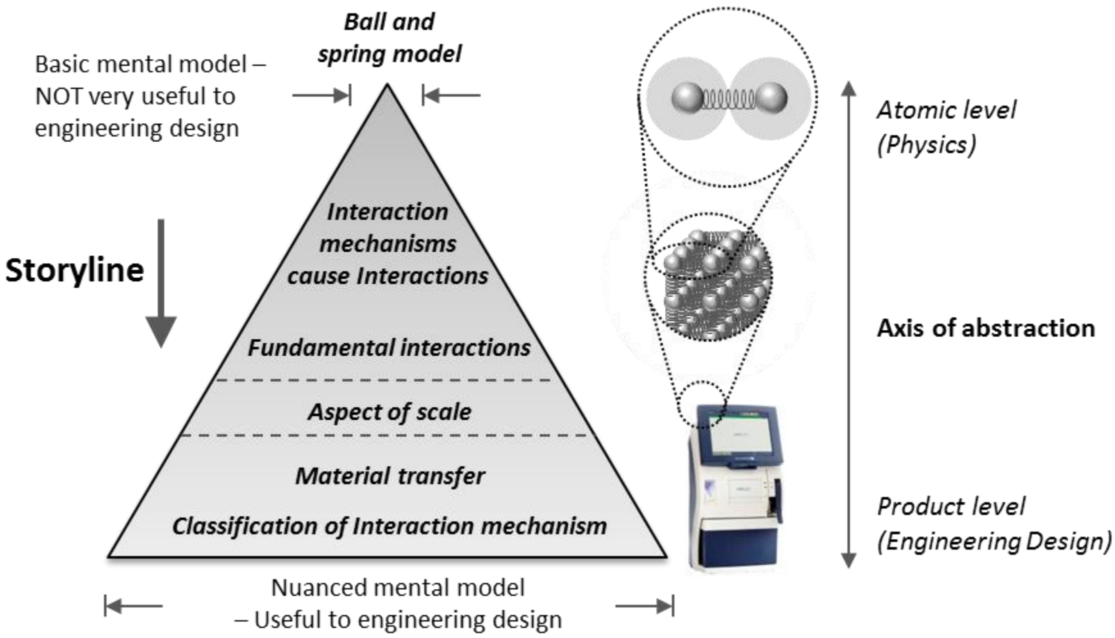


Fig. 1 Model showing the line of reasoning of the following section and how we progress from fundamental interactions to an interaction classification, which is useful to engineering design

An important objective of the framework is to equip the reader with the capacity to reason freely about interactions across the different length scales and variations with time. Understanding how interactions as described by various engineering disciplines are correlated is a core realization and take-away from this framework.

In order to set the stage for speaking about fundamental physics, we briefly introduce the notion of systems.

2.1 WHAT IS A SYSTEM?

A SYSTEM is, in its broadest form, a term used to articulate a collection of elements and their relations. Any relation may be considered as an input to or an output from a given system. The notion of a system is RECURSIVE meaning that the system has elements which themselves can be considered as systems with inputs and outputs. Systems theory is widely used in natural sciences to describe nature and physical 'things'. Anything can be considered as a system, from a smartphone to the universe. No matter what scale is considered, the same principles as explained above applies.
In physics, a SYSTEM is comprised of matter, which consists of any momentum-having particle or collection of particles that interact together and with its environment. From here on we use the term SYSTEM from this physical perspective unless otherwise stated.

2.2 LAWS OF CONSERVATION OF MOMENTUM AND ENERGY

A SYSTEM possesses *conserved properties* such as *translational momentum*, *angular momentum*, and *energy*. According to the *laws of conservation*, any *momentum* or *energy* which is gained by a given system is lost by its environment and vice versa. See below.

$\Delta\vec{p}_{sys} + \Delta\vec{p}_{surr} = \vec{0}$ (Law of conservation of translational (T) momentum)

$\Delta\vec{L}_{A,sys} + \Delta\vec{L}_{A,surr} = \vec{0}$ (Law of conservation of angular (A) momentum)

$\Delta\mathrm{E}_{sys} + \Delta\mathrm{E}_{surr} = 0$ (Law of conservation of energy)

(Chabay and Sherwood 2011)

None of these properties can disappear or appear from nothing because of conservation, although energy can be converted between different types of energy, e.g. an incandescent light bulb converts electric energy into electromagnetic radiation (radiative heat and light) and thermal energy with zero total loss.

2.3 SYSTEM BOUNDARY

A system is confined by its *system boundary* which conceptually separates the system from its environment. Therefore, what is not part of the system is part of its environment. A clear definition of the system boundary is instrumental to understanding the *state of a system*.

2.4 STATE OF A SYSTEM

It is useful in this context to consider *state of a system* to be the *sum of its conserved properties*; momentum (T & A) and energy. The only way to change the state of a system is by changing its conserved properties.

Because of relativity, one must declare a Frame of Reference (FoR) for the system boundary in order to model the properties of the system. This is analogous to the need for coordinate systems to position objects in space in mechanical engineering and a zero ground potential for electrical measurements in electrical engineering.

For simplicity reasons we assume zero acceleration in this framework, which means that the transition phase between the states is not part of this framework.

2.5 INTERACTIONS CAUSES SYSTEM STATE CHANGES

The instrument for affecting the conserved properties, and therefore the system state, is called INTERACTIONS.

An INTERACTION is equal to the transfer of momentum (T & A) and energy across a system boundary.

While INTERACTIONS cause a system's state to change, they must be facilitated by some physical phenomenon. We call this phenomenon the INTERACTION MECHANISM.

2.6 INTERACTION MECHANISM CAUSES INTERACTIONS

We therefore distinguish between two key concepts here; INTERACTION MECHANISM (cause) and INTERACTION (effect).

From physics it is known that, the transfer rate of momentum is equal to force (Chabay and Sherwood 2011).

$$Force = \frac{d(momentum)}{d(t)}$$

Therefore, as far as physics is concerned, we may deduce that the INTERACTION MECHANISM for facilitating INTERACTIONS is FORCE.

2.7 WHAT CONSTITUTES A FORCE?

Physics states that there are four fundamental interaction (FI) forces of nature; Gravitational, Electromagnetic, Strong, and Weak forces (Chabay and Sherwood 2011). All four types of force are considered as fields that radiate from a center point. The following table shows some of the key differences between the four fundamental forces such as range, their relative strength and what they affect, see table 2.

Table 2 The rows represent the four fundamental interaction forces of nature and their characteristics. The columns show key differentiators. The gravitational and electromagnetic forces are responsible for most physical phenomena in the engineering design of a product (Chabay and Sherwood 2011)

Name	Range (m)	Description
Gravitational	∞	Acts on mass. Always attractive.
Electromagnetic	∞	Acts on electrically charged particles. Can be either attractive or repulsive depending on the charge.
Weak nuclear	10^{-18}	Responsible for neutron decay
Strong nuclear	10^{-15}	Responsible for nucleus stability

All physical phenomena above the level of nuclear reactions can be explained using only *gravitational* and *electromagnetic forces*. Because engineering design mostly deals with a level above nuclear, the Strong and Weak forces are scoped out of this framework.

We will now briefly introduce some of the key characteristics of the gravitational and electromagnetic forces in order to stress just how fundamental and important these forces are to natural behavior of systems across any technical discipline.

2.7.1 Gravitational force

The gravitational force act between all objects having *mass* and is reciprocal, i.e. it acts equally and oppositely on both objects and is always attractive. It has infinite range and strength proportional to the inverse of the square of the distance between the objects. Therefore, all objects with mass in the universe act on all other objects of mass to a greater or lesser degree.

Gravitational field is a model used to explain the gravitational effects of individual objects by describing the gravitational force vector at a given point in space acting on a mass at that point. Gravitational forces can be constant or be varying; variation can only arise from either a) changing mass, or b) changing distance between the masses. We do not consider gravitational waves in this framework.

2.7.2 Electromagnetic force

Electromagnetic forces act between objects having *charge* (e.g. electrons, protons, ions, charged matter etc.), and is reciprocal (i.e. it acts equally and oppositely on the two objects). While all objects have mass, they do not necessarily have charge. The electromagnetic force can either be attracting or repelling, depending on the sign of the charges. It has infinite range and strength proportional to the inverse of the square of the distance between the objects. Therefore, all objects with charge in the universe act on all other objects with charge to a greater or lesser degree.

The Electromagnetic field is a model used to explain the forces that a charged object would exert at a given point in space if another charged object were placed there.

However, Electromagnetic fields are more complicated than Gravitational fields. Electromagnetic fields consist of two interrelated effects: electric fields and magnetic fields; the latter occurs when charges are moving relative to a frame of reference or spinning.

Electromagnetic fields can be constant or varying with time. Waves in Electromagnetic fields are called Electromagnetic radiation (EMR). These waves are caused by acceleration of the charge and are able to transmit momentum and energy, e.g. EMR is responsible for phenomena such as light.

Electromagnetic force is much stronger than Gravitational force when comparing elementary particles. It is only the size of the Earth combined with the fact that most materials are neutrally charged (i.e. have no net charge) that causes people to ”notice” gravitational effects as more significant than electromagnetic effects. However, Electromagnetic forces are responsible for the majority of systems' behavior and interactions between systems, due to electromagnetic field interactions at the atomic scale.

In order to understand how the gravitational and electromagnetic forces contribute to 'everyday' physical phenomena we present a useful mental 'ball-and-spring' model (Chabay and Sherwood 2011).

2.8 BALL-AND-SPRING MODEL

The model illustrates two atoms bound together by chemical bonds, here illustrated using a mechanical spring because of its analogous behavior. See fig. 2. A chemical bond means that two or more atoms are held together through reciprocal attraction of a shared number of electrons. The type of force responsible for this interatomic “spring-like” attractive effect is the electromagnetic force as described in fundamental physics. Because the two atoms have mass, they attract each other through the gravitational force as well. At this scale the gravitational force is however insignificant compared to the electromagnetic force.

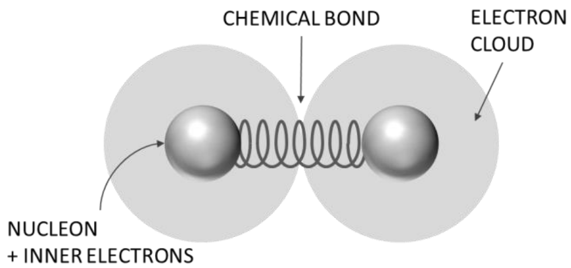


Fig. 2 Ball-and-spring model. It illustrates two atoms (balls) bound together by a chemical bond (spring). Redrawn from (Chabay and Sherwood 2011), January 2016, courtesy of Wiley

By means of this interatomic FORCE (INTERACTION MECHANISM) one atom may transfer physical properties to the other such as momentum (T & A) and energy (INTERACTION). In other words, they INTERACT facilitated by an INTERACTION MECHANISM.

2.9 ASPECT OF ABSTRACTION AND SCALE

We have now established the understanding that an INTERACTION MECHANISM is FORCE (Electromagnetic and Gravitational) illustrated by a ball-and-spring model of a system at quantum physics level of scale. However because the engineering design domain considers products at a much higher level of scale we must expand the notion of INTERACTION MECHANISM to also capture MATERIAL transfers, e.g. ventilation from a laptop pc or flow of gasoline, air and exhaust in an internal combustion engine etc.
We can therefore conclude that for systems at a product design scale the INTERACTION MECHANISM consists of:

- FORCE and/or
- MATERIAL transfer,

Both mechanisms have the capability to facilitate INTERACTIONS of various kinds.

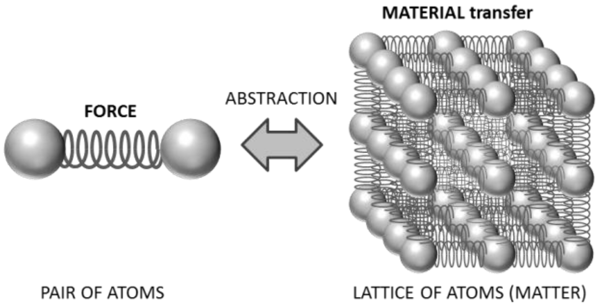


Fig. 3 Illustration of abstraction from a FORCE between two atoms to a lattice of atoms representing MATERIAL transfer

In order to be able to more easily distinguish between concepts, we introduce two classes of length scale which will be used throughout the framework:

- MICROSCALE (i.e. molecular scale and below etc.)
- MACROSCALE (i.e. larger than molecular scale etc.)

2.10 WHAT CONSTITUTES MATERIAL TRANSFER?

MATERIAL consists of matter, which can be considered as a network of atoms (i.e. balls) connected by chemical bonds (i.e. springs). See fig. 3, right. Because matter has *mass* and sometimes, but not always, *charge*, MATERIAL therefore also has *mass* and maybe *charge*. As a result of this, MATERIAL has a *gravitational field*, and sometimes an *electric/magnetic field* depending on its charge and movement. When the mass moves relative to a FoR it has *translational momentum*. If the mass is distributed (not just a point mass) and if it rotates with an angular velocity it has *angular momentum*. In addition MATERIAL has various forms of *energy* depending on the behavior of its constituent matter.

The significance of a physical property depends on the system's (e.g. a pair of atoms) behavior at an atomic scale, for example whether the atom pairs are rotating, spinning, oscillating, stretched, compressed, moving relative to a frame of reference etc.

We will now describe the different types of INTERACTIONS that are associated with the one INTERACTION MECHANISM called MATERIAL transfer.

2.10.1 INTERACTIONS facilitated by MATERIAL transfer

Translational momentum
Any MATERIAL has mass at rest. When the MATERIAL moves translationally with a certain velocity relative to a FoR, it has translational momentum. Changing a MATERIAL's momentum requires a net force exerted over a time interval. In fig. 4-1 this phenomenon is explained using a pair of particles (ball) although the principle can be scaled indefinitely. An important distinction between momentum and kinetic energy is the fact that momentum is a vector quantity, i.e. magnitude and direction.

Angular momentum

If the MATERIAL is exposed to a torque then the MATERIAL gains *angular momentum*. Angular momentum can further be subdivided into *rotational angular momentum*, which describe rotational motion around its own axis, or *translational angular momentum*, which describes the MATERIALs rotation around an external point of location. To put it differently, a MATERIAL may be spinning (rotational angular momentum) around itself and/or orbiting (translational angular momentum) around an external point, just like the earth orbiting the sun while spinning about its own axis. See right side of fig. 4-2.

An important modeling aspect about momentum is that it is a vector-quantity meaning it has direction and magnitude relative to a FoR. When determining the behavior of MATERIAL, it is the sum of all vector forces that result in a certain behavior.

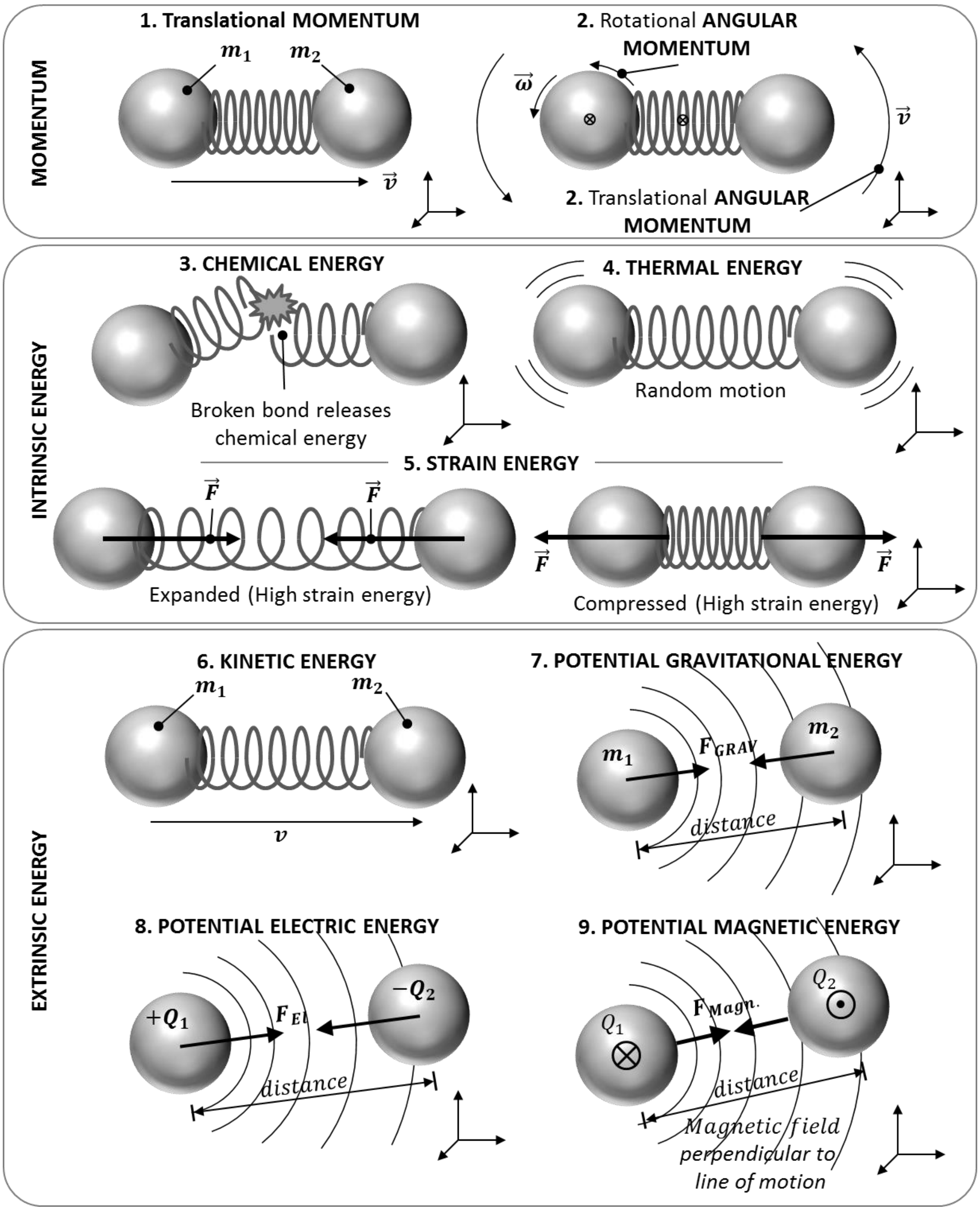


Fig. 4 Explanation of the various physical properties (INTERACTIONS) associated with MATERIAL transfer. They are divided into momentum, intrinsic and extrinsic energies

Mass transfer

Mass transfer rate represent the amount of mass being transferred per time unit. It is essentially already captured in the momentum equation, however it may be convenient to account for the mass transfer rate independently, because it is a useful measure in many applications, e.g. ensuring that the rate of mass flow (of water) being pumped out of the hull of a container ship counterbalances the on-loading of the cargo, ensuring that the rate of mass flowing into a gas turbine is equal to that flowing out of the turbine etc.

Charge transfer

Charge transfer rate is essentially a measure of the amount of charges that is transferred per time unit. In electrical engineering this is called *electric current*. The use of the wording *charge transfer rate* is a bit broader than electric current, in that it also includes charges at higher abstraction level such as ions. This becomes relevant in for example electrolysis or osmosis.

Energy

Energy is a concept used to describe physical behavior of nature. It is a concept that has evolved into many different classes of energy describing natural behavior at various levels of abstraction as scalar quantities. These abstractions make it easier to model and compare various physical phenomena across different technical disciplines without having to relate to the behavior of elementary particles. It is however the authors' opinion that understanding the physical phenomena at a basic level will improve the freedom to communicate with other technical disciplines outside one's own area of technical expertise.

The following subclasses of energy are arbitrarily chosen, although they seem to capture most physical phenomena that are addressed in engineering design of products. Nuclear energy describes the energy required to keep the atomic nucleus together. Including nuclear energy in this framework would have required us to also include the Strong force as a fundamental interaction, which is not considered as part of the scope for this framework.

The subclasses of energy are here divided into two groups; *intrinsic* and *extrinsic* energies. An *intrinsic energy* is an inherent property of the MATERIAL independent of its size and its surroundings, whereas an *extrinsic energy* is the result of the MATERIAL being under influence of some external field or relative to some other entity. Extrinsic energies therefore require an external Frame of Reference (FoR) to make sense.

Chemical energy (intrinsic) is a measure of how much energy is released or absorbed when interatomic bonds are broken or created. See fig. 4-3. Some MATERIALs such as gasoline, has a relatively high chemical energy compared to its mass which is exploited in internal combustion engines. When gasoline an air is compressed inside a chamber and a spark is lit, the air and gasoline will chemically react which results in a gaseous substance (i.e. exhaust) which causes a rapid expansion. This expansion is the utilized to create a mechanical movement to propel the vehicle. Other common examples are batteries, fuel cells electrolysis.

Thermal energy (intrinsic) means that the atoms inside the MATERIAL are vibrating in a random pattern, more or less active. See fig. 4-4. In other words each atom oscillates between two static end states. In its transition from one end state to another the atom has dynamic kinetic energy. As it reaches one of the end states and before it transitions to the other end state all of the kinetic energy has been converted into potential energy. Instead of measuring the kinetic and potential energy of each atom in a substance, a thermometer measuring the temperature is used as a proxy for the thermal energy. When the temperature reaches near the point of absolute zero, all atoms stops moving, meaning that the MATERIAL has zero thermal energy.

Strain energy (intrinsic) is a measure of the level of stress that a MATERIAL exhibits due to its level of expansion or compression. See fig. 4-5. As explained earlier, when two atoms are forced together they are compressed relative to its normal state. When compressed, the two atoms will experience an outward pointing force due to the electromagnetic force, which represents a strain in the MATERIAL. The opposite case also exists where two atoms are forced apart causing an inward pointing force, e.g. the stretching of rubber when blowing up a balloon.

As a consequence of the MATERIAL moving into the receiving system, it may exert a *force* due to its contained strain energy, e.g. if the MATERIAL is compressed. Such a force may need to be accounted for when designing certain products, e.g. when blowing compressed air into a tire, the valve needs to withstand not only the pressure of the flow but also the force caused by the sudden expansion of the compressed gas.

Kinetic energy (extrinsic) is a measure of a MATERIAL's motion energy. See fig. 4-6. It is a scalar quantity (in opposition to momentum which is vector-based) of MATERIAL moving at constant velocity relative to a FoR. When a MATERIAL is under influence of a force and changes velocity to a new constant velocity, that force has done *work* on the MATERIAL. The amount of work used to bring the MATERIAL to the desired velocity is equal to the (delta) kinetic energy of the moving MATERIAL (i.e. discounting energy losses due to friction).

Potential gravitational energy (extrinsic) is a measure of the amount of *work* (mechanical energy) that is needed to move MATERIAL a given distance relative to a FoR when influenced by a gravitational field. See fig. 4-7. For example propelling a shuttle out of the atmosphere of earth requires enough energy to overcome the gravitational force of the earth. The energy used to lift the shuttle is equal to the potential gravitational energy of the shuttle in space (i.e. neglecting the weight of the payload and friction of air). Albeit all masses attract each other through the gravitational force, it is predominantly that of the earth's that needs to be accounted for in product design.

Potential electric energy (extrinsic) is a measure of how much *work* (electric energy) is needed to move a charge influenced by an electric field (Energy/Charge = Voltage). See fig. 4-8. Say an electron (charge; 1,6*10^-19 C) moves through a potential difference of 1 Volt (1 J/C) then the magnitude of the electrons electric potential energy change is C * J/C = 1,6*10^-19 Joule. Examples are electric circuits, static electric objects.

Potential magnetic energy (extrinsic) is a measure of how much work (energy) is needed to move magnetized MATERIAL a given distance under influence of a magnetic field. See fig. 4-9. An example could be magnetic bike lights where a permanent magnet mounted on the spokes of the wheel, charges a capacitor in the bike light through electromagnetic induction.

2.11 SUMMARY

The following matrix sums up the relationship between INTERACTION MECHANISM (cause) and INTERACTION (effect), see table 3.

Table 3 Matrix showing the relationship between the two types of INTERACTION MECHANISMS and the possible INTERACTIONS

	Possible INTERACTIONS (Effect)	
INTERACTION MECHANISMS (Cause)	TRANSFER OF MOMENTUM (T&A)	TRANSFER OF ENERGY
FORCE	**YES** Via direct action.	**YES** Both intrinsic and extrinsic energies
MATERIAL TRANSFER	**YES** Via addition of moving mass.	**YES** Both intrinsic and extrinsic energies

2.12 CLASSIFICATION OF INTERACTION MECHANISM

The following section will systematically classify the various forms of INTERACTION MECHANISMS. For every class of INTERACTION MECHANISM, the relevant INTERACTIONS will thus be listed to expose the distinction between the two concepts.

As introduced earlier there are in general two INTERACTION MECHANISMs that facilitate an interaction.

- FORCE at system boundary (Zero MATERIAL transfer)
- Net MATERIAL transfer across a system boundary

In other words, if there is *no* FORCE and *no* net MATERIAL transfer between two systems, then there is *no* interaction and the systems are unchanged. If there on the other hand is a FORCE and/or a MATERIAL transfer (INTERACTION MECHANISMS), then one or more physical properties (INTERACTIONS) may be transferred and thus change or sustain the state of the systems. The aspect that determines whether an INTERACTION MECHANISM changes or sustains the state of a system depends on the behavioral characteristics of the mechanism.

What do we mean by NET interaction?
When considering a system as a whole all forces acting on the system and all transfers of momentum to and from the system must be balanced out (vector sum), because of our decision to exclude acceleration from this framework. However, the same does not apply for MATERIAL transfer which may facilitate an INTERACTION. Thus when summing up all of the MATERIAL flow in and out of a system, there may be an excess of MATERIAL flowing into the system compared to that flowing out leading to a NET INTERACTION flow that *changes the system state*. Hence, NET MATERIAL transfer leads NET INTERACTION, which causes the *system to change state*.

However, INTERACTIONS may also act to *sustain the state of a system*, meaning that the NET INTERACTION must be zero, i.e. and the total energy entering and leaving the system is zero (scalar sum of different forms of energy).

2.12.1 FORCE at system boundary (Zero MATERIAL transfer)

At the ”everyday” scale in which product design takes place, the effects of Electromagnetic forces at different length scales are experienced very differently. Electromagnetic forces acting at the MICROSCALE between elementary particles are experienced at a MACROSCALE as properties of the bulk MATERIAL; solids, liquids, gases, hardness, contact etc. Electromagnetic forces acting between modules or product at a MACROSCALE are experienced as they truly are: attractions between charged objects, magnetic effects and electromagnetic radiation.

Therefore, it may first of all be convenient to reason about FORCE at different levels of scale, e.g. at MICRO- and MACROSCALE. The first level of decomposition is therefore guided by a *length scale* perspective.

In order to capture the behavioral aspects of the INTERACTION MECHANISM and to link them to the associated INTERACTIONS, we may further decompose MICRO- and MACROSCALE FIELD EFFECTS according to *patterns of movement*. As such we find the following subclasses to be useful in capturing all FORCE-initiated physical phenomena, see table 4.

Table 4 Classification of INTERACTION MECHANISMS

INTERACTION MECHANISMS

TYPE OF FORCE	LENGTH SCALE	PATTERN OF MOVEMENT
ELECTROMAGNETIC FIELD FORCE	MICROSCALE FIELD EFFECTS (PHYSICAL CONTACT)	RANDOM collision forces (Asynchronous oscillations)
		STATIC force (Synchronized displacement)
		CONSTANT MOVING force (Synchronized displacement)
		WAVES (Synchronized oscillations)
	MACROSCALE	STATIC force

FIELD EFFECTS		
		CONSTANT MOVING force
		WAVES propagating in electromagnetic field
GRAVITATIONAL FIELD FORCE		STATIC force
		CONSTANT MOVING force

We will now describe each class of INTERACTION MECHANISM in depth and address the associated INTERACTIONS that are facilitated by the respective INTERACTION MECHANISMS.

ELECTROMAGNETIC MICROSCALE FIELD EFFECTS

The notion of *physical contact* represents the case where a repelling FORCE arises between two systems' surfaces when brought in close proximity of each other. In other words, the two systems physically "touch" at a system boundary when experienced at a product scale. Any of the following INTERACTIONS will always be facilitated when physical contact although the significance depends on the movement patterns of the atoms.

RANDOM collision force - MICROSCALE EM FIELD FORCE EFFECT (PHYSICAL CONTACT)

Conduction of heat is the phenomenon where thermal energy is transferred from the "hot" system to the "cold" system without MATERIAL transfer until both systems have reached equal thermal energy levels (equilibrium). The energy (and momentum) is transferred through interatomic collisions between the heavily oscillating atoms in the "hot" system and the less oscillating atoms in the 'cold' system. Through these collisions, momentum and kinetic energy is transferred from the 'active' to the 'less active' atoms. See fig. 5. The bigger the difference in temperature between the two systems, the higher the rate of thermal energy transfer.

We may characterize this RANDOM pattern of atomic motion as ASYNCHRONOUS OSCILLATIONS of MICROSCALE AMPLITUDE of atoms comprising contact surfaces with ZERO NET FORCE.

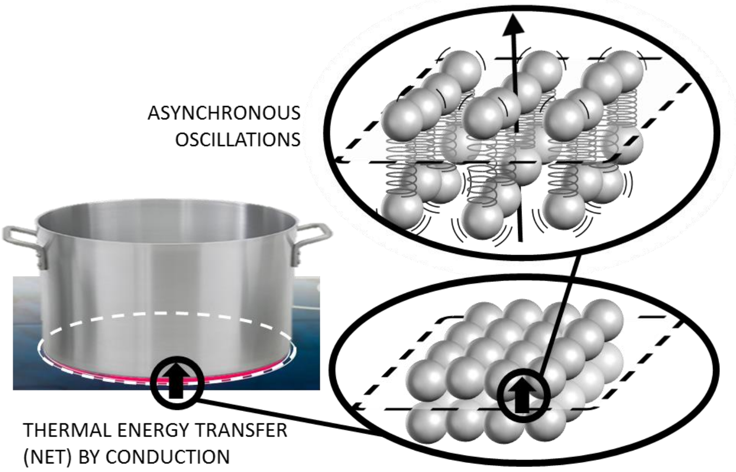


Fig. 5 Illustration shows a pot on a ceramic stove. By gradual zooming in, we expose random interatomic collision FORCES at the system boundary (right), which causes transfer of kinetic energy from atom to atom. Due to abstraction this is modeled as THERMAL ENERGY TRANSFER by conduction (left). *'Pot on stove'* image, courtesy of Adobe Stock

Because it is not useful consider the momentum and kinetic energy transfer of each and every interatomic collision and then sum it up, *thermal energy transfer* is used as an abstraction for the INTERACTION.

If the input heat transfer is equal to the output heat transfer, of a particular system, the *state is sustained*. Also, if the two systems have the same thermal energy at the contact surface, there will not be a heat transfer and the *system state will be sustained*.

When two systems are in physical contact at a system boundary and there is a difference in temperature the following INTERACTION takes place:

- Thermal energy transfer

STATIC force - MICROSCALE EM FIELD FORCE EFFECT (PHYSICAL CONTACT)

A *contact force* is an abstraction because it represents the mean force acting between billions of atoms on each surface of the contact. The fact that it is *static* relates to the behavior of the force. It does thus not change over time or displace relative to a FoR, e.g. a constant force acting in an assembly interface.

A static force does not in itself lead to any momentum (T & A) or energy transfer across the system boundary however it may be responsible for SUSTAINING the energy state of a system, e.g. the force from a mounting on a bookshelf prevents it from falling to the ground thus losing its potential gravitational energy. Thus without the static force the system will change its state.

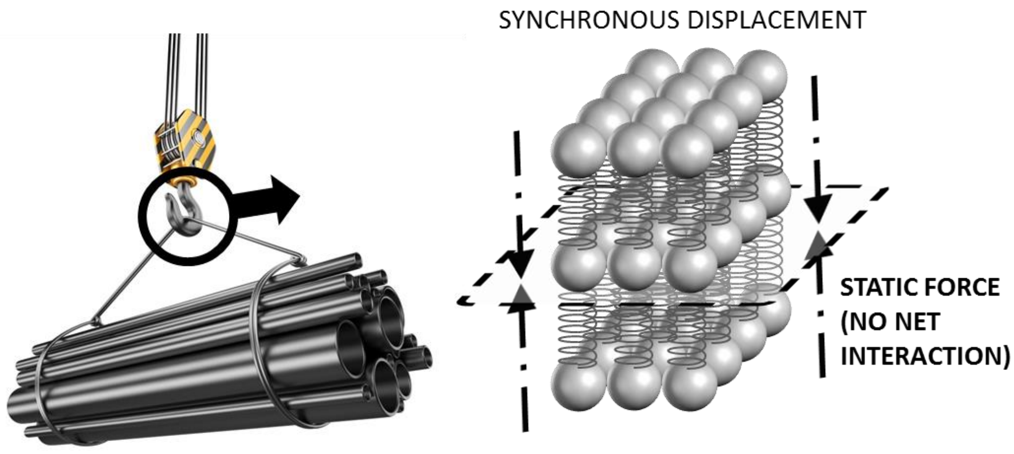


Fig. 6 Illustration shows how a static force (INTERACTION MECHANISM) is sustaining the state of the system. It is assumed that the cargo hangs with a constant distance to the ground meaning that the system is static in the ground's FoR as well as the cargo's. *'Crane lifting steel'* image, courtesy of Adobe Stock

We may characterize this *static* contact force as a SYNCHRONIZED DISPLACEMENT of atoms comprising one contact surface against those comprising the other contact surface, relative to a FoR. The contact surfaces hence displace relative to their respective systems FoR, but not relative to each other resulting in a balanced net force with zero rate of change (constant force) at the system boundary. See fig. 6.

CONSTANT MOVING force – MICROSCALE EM FIELD FORCE EFFECT (PHYSICAL CONTACT)

When two systems have physical contact, there may be a net force acting between the systems causing the point of force application to move with constant velocity either because the affected systems compress/expand or move (translational or rotational) relative to a FoR. The force is thus displaced a certain distance relative to a FoR thus performing *work* (transfer of strain energy) from one system to the other. See fig. 7.

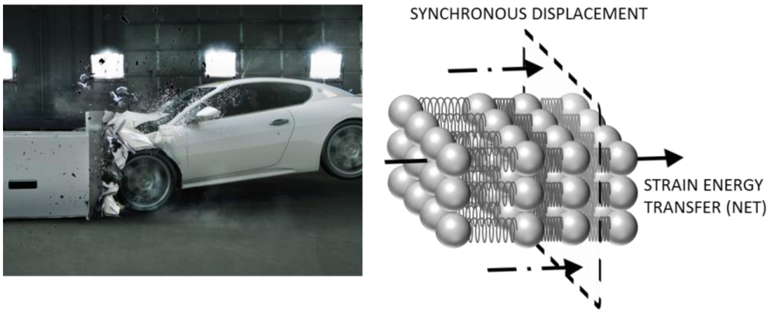


Fig. 7 Illustration shows how a car's kinetic energy is converted to strain energy upon collision with the barrier. The barrier transfers strain energy to the car (in the car's FoR) by compressing it. *'Crash test'* image, courtesy of Adobe Stock

There is constant force acting between the barrier and the car which leads to a constant deceleration over time.

When two systems are in physical contact at a system boundary and there is a net force acting, the affected system might be compressed meaning that *strain energy* is transferred or it may be displaced at constant speed relative to a FoR. In this framework, both scenarios are categorized as strain energy transfer. The following INTERACTIONS are facilitated by a constant, moving FORCE:

- Momentum (T & A)
- Intrinsic energy
    - Strain energy (i.e. displacement of force due to compression/expansion or bulk movement)

WAVES – MICROSCALE EM FIELD FORCE EFFECT (PHYSICAL CONTACT)

If the magnitude of a force varies continuously with a given frequency, it elastically compresses the affected system thus transferring strain energy in an oscillating way. This transfer of energy propagates as a wave through the affected system without any MATERIAL transfer, e.g. sound waves, seismic vibrations etc.

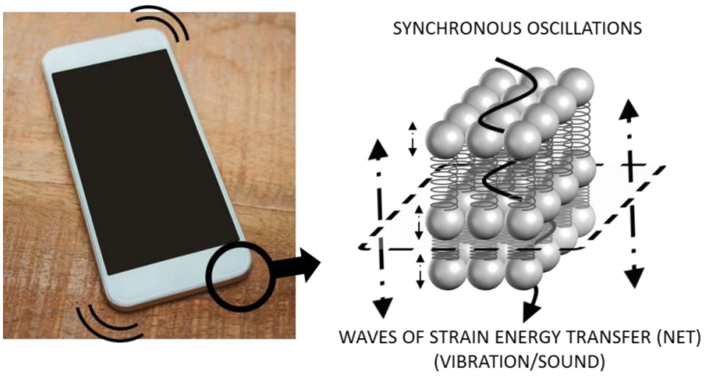


Fig. 8 Illustration shows how a vibrating smartphone transfers waves of strain energy into the table through synchronized oscillations of atoms. The frequency of the waves determines whether they can be heard by humans as sound. *'Vibrating smartphone'* image, courtesy of Adobe Stock

We may consider this as SYNCHRONIZED OSCILLATION of atoms comprising one contact surface causing those of the other contact surface to move in sympathy, resulting in an WAVE-like FORCE at the system boundary. See fig. 8.

When two systems are in physical contact at a system boundary and there is a WAVE-like net FORCE acting, waves of *strain energy* propagate across the system boundary and through the affected system. The characteristics of the movement of the contact force determine the *rate of strain energy transfer*.

- Transfer of momentum
- Strain energy transfer (i.e. sound/vibrational energy)

The following table lists examples of INTERACTIONS facilitated by a MICROSCALE ELECTROMAGNETIC (EM) FIELD FORCE EFFECT (INTERACTION MECHANISM), see table 5.

Table 5 Examples of INTERACTIONS facilitated by MICROSCALE FORCE (INTERACTION MECHANISMS)

EXAMPLES	INTERACTION MECHANISM (CAUSE)	INTERACTION (EFFECT)
Cup on a table	MICROSCALE EM FIELD EFFECTS - *STATIC*	The static contact FORCE does not facilitate any INTERACTION, although it serves the purpose of SUSTAINING the state of the systems. Without the static FORCE between the table and the cup, the cup would fall to the ground due to the Earth's gravitational field.
Heat sink	MICROSCALE EM FIELD EFFECTS - *RANDOM*	Heat from a CPU is removed through heat conduction to a metal heat sink, which has a large surface area. When air is passed by the heat sink, heat is conducted to the air across that surface area.
Pedal (when biking)	MICROSCALE EM FIELD EFFECTS – *CONSTANT MOVING*	A person's foot does work on the pedal by forcing it around and axis point thus transferring strain energy. This energy is through a series of conversions directed to the asphalt causing the bike to move faster (gain kinetic energy).
Loudspeaker	MICROSCALE EM FIELD EFFECTS – *WAVES*	Vibrating membrane of loudspeaker transmits waves of strain energy (i.e. sound) into the surrounding air.
Pick up on a phonograph	MICROSCALE EM FIELD EFFECTS – *WAVES*	Rotation of the record combined with the variation of groves in the record transfers waves of strain energy to the pickup, which then converts it into electrical signals that can be amplified and played as sound through loudspeakers.

ELECTROMAGNETIC (EM) MACROSCALE FIELD EFFECTS

The notion of MACROSCALE EM FIELD FORCE EFFECTS constitutes the case where a charged system is affected by its surrounding through Electromagnetic forces, which may be modeled as a field. One of the differences from the physical contact is the fact that the Electromagnetic force affects the charged system as a whole.

STATIC EM FIELD FORCE EFFECTS - MACROSCALE

When two systems affect each other through a *static* field force, and thus does not move relative to each other, then there is NO INTERACTION taking place – only force as an INTERACTION MECHANISM.

CONSTANT MOVING EM FIELD FORCE EFFECTS - MACROSCALE

When the charged systems move with constant speed relative to a FoR they generate both electric fields and magnetic fields – hence electromagnetic fields. The only way of increasing or decreasing the magnitude of the force is by increasing the charge or a system or by minimizing the distance between the charged systems and vice versa.

When two systems affect each other through a CONSTANT MOVING field force, it facilitates the transfer of INTERACTIONS of the kind:

- Momentum (T & A)
- Extrinsic Energy
    - Kinetic energy
    - Potential electric energy
    - Potential magnetic energy

WAVES propagating in EM field force - MACROSCALE

When charges are accelerated both the electric and magnetic fields are momentarily disturbed creating a wave/particle-like effect that radiate outward through space from the center of the accelerated charge. These disturbances in the field carry with them energy and momentum (T&A).

If a wave reaches a distance far enough from the accelerated charge, the wave may continue to propagate or radiate freely, independent of the charge that created it. These waves are typically referred to as *propagating, far-field electromagnetic waves*, e.g. light, radio waves etc. On the other hand, *non-propagating, near-field electromagnetic waves* rely on the continuous oscillating accelerations of charges in a circuit to sustain its presence, e.g. electromagnetic induction. Collectively, these physical phenomena are called *electromagnetic radiation (EMR)*. It is not the purpose of this paper to discuss this duality-aspect of EMR further, however we will from here on treat EMR as FIELD EFFECTS, and not as MATERIAL transfer.

Unlike sound, which is generated by a vibrating electromagnetic contact FORCE, EMR does not require a "medium" or MATERIAL to propagate through, e.g. light travels from the sun to the earth in a vacuum.

When two systems affect each other through a near- or far-field electromagnetic wave, they facilitate the transfer of INTERACTIONS of the kind:

- Momentum (T & A)
- Extrinsic Energy
  - Kinetic energy
  - Potential electric energy
  - Potential magnetic energy

The rate of energy transfer depends on the frequency of the wave, i.e. low-frequency/low-energy radio waves, through visible light, to high-frequency/high-energy gamma radiation.

GRAVITATIONAL (GRAV) FIELD EFFECTS

The notion of gravitational FIELD EFFECTS constitutes the case where a system, which is by definition mass-having, is affected by its surrounding through gravitational forces, which may be modeled as a field. One of the differences from the electromagnetic fields is the fact that it is always attractive and acts only on mass. The gravitational force affects the system as a whole.

STATIC GRAV FIELD FORCE EFFECT

Two systems may be sustaining their state, if the distance between them remains constant over time and the magnitude of the masses also remains constant over time.
When two systems affect each other through a static field force, and thus does not move relative to each other, then there is no INTERACTION – only force as an INTERACTION MECHANISM.

CONSTANT MOVING *GRAV FIELD FORCE EFFECT*

If the distance between two systems increase or decrease or the masses change, then the gravitational force changes.

When two systems affect each other through a CONSTANT MOVING field force, they facilitate the transfer of INTERACTIONS of the kind:

- Momentum (T & A)
- Extrinsic Energy
  - Kinetic energy
  - Potential gravitational energy

The following table lists examples of INTERACTIONS facilitated by FIELD FORCE based INTERACTION MECHANISMS, see table 6.

Table 6 Examples of INTERACTIONS facilitated by FIELD FORCE (INTERACTION MECHANISMS)

EXAMPLES	**INTERACTION MECHANISM (CAUSE)**	**INTERACTION (EFFECT)**
A skydiver	*CONSTANT MOVING* - GRAV FIELD FORCE EFFECTS	A skydiver “falls” to the ground because the earth (vast mass) and the skydiver (small mass) performs work on each other through a significant gravitational force field
Microwave oven	*WAVES* - MACROSCALE EM FIELD FORCE EFFECTS (i.e. EMR/microwave)	An electric current is converted to an electromagnetic field (i.e. microwave range) through a magnetron, which causes water molecules and fat to oscillate and thus heat up the food.
Smartphone display	*WAVES* - MACROSCALE EM FIELD FORCE EFFECTS (i.e. EMR/visual spectrum)	Light emitted from the screen to the user.
Toaster	*WAVES* - MACROSCALE EM FIELD FORCE EFFECTS (i.e. EMR/broad spectrum)	Electromagnetic radiation is radiated from a “hot” wire and visible to the human eye as a red glow.
Inductive charging	*WAVES* - MACROSCALE EM FIELD FORCE EFFECTS (i.e. EMR, near-field)	An alternating electric current in a charger produces an alternating magnetic field which “wirelessly” interacts with an electric generator that inverts the process from alternating magnetic field to electric current thus charging a battery.
A FM-radio	*WAVES* - MACROSCALE EM FIELD FORCE EFFECTS (i.e. EMR/radio waves)	Radio stations emit radio waves in a specific pattern. The waves carry with them energy and momentum which can be converted to electrical signals and sound through a radio.
Magnetically levitating trains	*STATIC* - MACROSCALE EM FIELD FORCE EFFECTS	Certain high-speed bullet trains levitate above the tracks through magnetic repulsion between the train and the tracks generated by electromagnets.

2.12.2 Net MATERIAL transfer across a system boundary

A net MATERIAL transfer is an INTERACTION MECHANISM where MATERIAL is transferred across a system boundary, thus carrying with it various physical properties which constitute the INTERACTION.

MATERIAL at different scales

Any momentum-having matter is considered as MATERIAL transfer. From an engineering design perspective, this is not very convenient because MATERIAL transfer potentially covers everything from electrons flowing in a circuit to wind propelling a windmill. We therefore classify MATERIAL transfer into two classes based on length scale:

- ELEMENTARY PARTICLES (i.e. captures MATERIAL at a MICROSCALE)
- BULK MATERIAL (i.e. captures MATERIAL at a MACROSCALE)

This classification is not absolute, but merely serves the purpose of supporting a mental division of concepts by physical length scale. Examples of ELEMENTARY PARTICLE MATERIAL transfer are electricity, electrolysis, osmosis etc. Examples of BULK MATERIAL transfer are hydraulics, pneumatics, advection, diffusion etc.

As indicated by the examples of BULK MATERIAL, it is evident that MATERIAL may be in different phases:

- Solids (typically maintains a fixed volume and shape)
- Liquids (typically maintains a fixed volume with amorphous shape)
- Gases (no fixed volume or shape)
- Plasma (no fixed volume or shape, free moving ions, electrons)

While these different phases seem rather distinct and non-compatible they do mix and co-exist, e.g. oxygen and carbon-dioxide (gas) in human blood (liquid), ice (solids) in water (liquids).

It does not seem useful to further classify ELEMENTARY PARTICLES and BULK MATERIAL transfer according to characteristics of their pattern of movement as was done with the FORCE-based INTERACTION MECHANISM.

Instead we may refer to section 2.10.1 for information on what INTERACTIONS are facilitated using MATERIAL transfer as INTERACTION MECHANISM.

For MATERIAL transfer on a MICROSCALE (i.e. ELEMENTARY PARTICLES) some types of energy may be less relevant because they describe the properties of MATERIAL at a higher level of abstraction than freely moving ELEMENTARY PARTICLES. These types are; chemical energy, thermal energy, and strain energy.

The following table lists common examples INTERACTIONS via net MATERIAL transfer. The list of related INTERACTIONS only exposes those of greatest interest. See table 7:

Table 7 Examples of technical systems that apply MATERIAL transfer to facilitate the transfer of momentum and/or energy

EXAMPLES	**INTERACTION MECHANISM (CAUSE)**	**INTERACTION (EFFECT)**
Hydraulic fluid systems	BULK MATERIAL (MACROSCALE)	Incompressible hydraulic fluid (MATERIAL) that transfer momentum and energy.
Pneumatic system	BULK MATERIAL (MACROSCALE)	Gaseous fluid (MATERIAL) that transmits strain energy due to its compressed nature in pneumatic systems.
Bike brake system (cable)	BULK MATERIAL (MACROSCALE)	A solid (MATERIAL) transmitting force. It passes from one system to another once activated carrying energy.
Electricity	ELEMENTARY PARTICLES (MICROSCALE)	Charged particles (MATERIAL) flow from source to sink. Charges can essentially also be transferred with BULK movement of matter. Charges carry momentum and energy.
Gasoline car	BULK MATERIAL (MACROSCALE)	Gasoline is a liquid with high chemical energy used to fuel combustion engines in cars.
Potatoes in water (osmosis)	ELEMENTARY PARTICLES (MICROSCALE)	Water molecules dissipate through a permeable membrane to neutralize differences in salt concentration levels. In general this is transfer of ions (charged molecules). The molecules carry energy.
Propulsion system in space	BULK MATERIAL (MACROSCALE)	Gas is ejected at high velocity with *momentum* which creates a force. Due to conservation of momentum, a reaction force causes the spacecraft to accelerate.
A pump	BULK MATERIAL (MACROSCALE)	A pump performs work on a fluid when forcing it through a circuit. The fluid carries momentum and energy.

2.13 SUMMARY OF THE INTERACTION FRAMEWORK

The proposed *Interaction Framework* presents the classification of INTERACTION MECHANISM, which facilitates the concept of INTERACTION. The INTERACTION MECHANISM classification is characterized by being *mutually exclusive* (no overlap) and *collectively exhaustive* (i.e. covers all technical disciplines that are relevant to engineering design). In other words, any physical phenomenon, indifferent from technical discipline, may only fall into *one* of the eleven detailed classes of INTERACTION MECHANISMS.

The following matrix summarizes the framework, see table 8; both the classification of INTERACTION MECHANISMS and INTERACTIONS as well as the relations between them. It should be mentioned that when choosing an INTERACTION MECHANISM, not all INTERACTIONS may necessarily be of equal importance to specify. In fact some might be totally omitted from the documentation due to their insignificant influence in the given design situation. However, by being confronted with all possible INTERACTIONS and INTERACTION MECHANISMS, we limit the risk of missing detrimental INTERACTIONS. We leave it up to the system architect to decide what to specify in the specific situation.

Table 8 Classification of INTERACTION MECHANISMS and INTERACTIONS and how they relate. The table is equal to table 3 except for the greyed out fields, which are further classifications

						INTERACTION (EFFECT)	
	PRIMARY (ABSTRACTION)	SECONDARY (TYPE)	TERTIARY (LENGTH SCALE)	QUATERNARY (PATTERN OF MOVEMENT)	EXAMPLES USING FAMILIAR DOMAINS	**TRANSFER OF MOMENTUM** - (TRANSLATIONAL & ANGULAR)	**TRANSFER OF ENERGY**
INTERACTION MECHANISM (CAUSE)	**FORCE**	ELECTRO-MAGNETIC	MICROSCALE FIELD EFFECTS (PHYSICAL CONTACT)	RANDOM	*Thermal conductivity, stove, radiator etc.*	*NO*	THERMAL
				STATIC	*Assembly interfaces*	*NO*	*NO*
				CONSTANT MOVING	*Crane lifting container, compression of material, rotating shaft etc.*	YES	STRAIN
				WAVES	*Pistons, sound, earthquakes etc.*	YES	STRAIN, KINETIC
			MACROSCALE FIELD EFFECTS	STATIC	*Balloon on a jumper, permanent magnet/ electromagnet*	*NO*	*NO*
				CONSTANT MOVING	*Solenoid (constant current increase assumed)*	YES	ELECTRIC POT., MAGNETIC POT.
				WAVES	*EMR (i.e. sunlight, x-rays, UV-light, induction etc.)*	YES	KINETIC, ELECTRIC POT., MAGNETIC POT.
		GRAVITATIONAL	-	STATIC	*Earth's field (approx.)*	*NO*	*NO*
				CONSTANT MOVING	*Black holes with constant mass gain*	YES	GRAVITATIONAL POT.
	MATERIAL TRANSFER	-	ELEMENTARY PARTICLES (MICROSCALE)	CONSTANT MOVING	*Electricity, electrolysis, osmosis, diffusion etc.*	YES	KINETIC, GRAVITATIONAL POT., ELECTRIC POT., MAGNETIC POT.
			BULK MATERIAL (MACROSCALE)	CONSTANT MOVING	*Hydraulics, pneumatics, advection, etc.*	YES	CHEMICAL, THERMAL, STRAIN, KINETIC, GRAVITATIONAL POT., ELECTRIC POT., MAGNETIC POT.

2.13.1 Principle of Superimposition

In order to also capture even sophisticated physical phenomena in which several INTERACTION MECHANISMS happen simultaneously, we include a *Principle of Superimposition*. This means that while the different types of INTERACTION MECHANISMS do not overlap from a physics perspective, they may be superimposed to describe a certain physical phenomenon, e.g. when transferring strain energy via a contact FORCE at a boundary, you may simultaneously conduct heat or when transferring MATERIAL at a MACROSCALE level of scale you may simultaneously transmit electric charges through the MATERIAL at a MICROSCALE. Utilizing this principle will allow the system architect to reason about very complex INTERACTIONS and decompose them into the various classes for later specification.

Below is a list of examples of superimposed INTERACTION MECHANISMS explained using the framework, see table 9.

Table 9 Everyday examples explained using superimposed INTERACTION MECHANISMS

EVERYDAY EXAMPLES	SUPERIMPOSED INTERACTION MECHANISMS	INTERACTIONS (those relevant)
Blowdryer in public restrooms	A hot air stream (BULK MATERIAL) dries the users hands while emitting UV-light (MACROSCALE, EM FIELD FORCE - WAVE) to kill bacteria	The air stream facilitates mass, momentum, thermal energy, and kinetic energy transfer. The light facilitates momentum and kinetic energy transfer.
Ionic hair dryers	A hot air stream (BULK MATERIAL) dries the user's hair while emitting electrons (ELECMENTARY PARTICLES) that neutralizes static charged hair.	The air stream facilitates mass, momentum, thermal energy, and kinetic energy transfer. The electrons facilitate charge, momentum, kinetic energy, and electric potential energy transfer.
An auto air conditioning system with built-in scent	Outside air (BULK MATERIAL) is cooled in the AC system of a car and infused with certain scents (ELEMENTARY PARTICLES) that freshen the ambient air in the cabin	The air conditioning stream (MATERIAL) facilitates mass, momentum, thermal energy, and kinetic energy transfer. The scent facilitates chemical energy, momentum, and kinetic energy.
Fountain with fiber optics in water beams	Water (BULK MATERIAL) is ejected from a nozzle. In addition, there is a fiber optic cable emitting light (MACROSCALE, EM FIELD FORCE - WAVE), through the water beam	The water (MATERIAL) carries with it mass, momentum, kinetic energy, and potential gravitational energy. The light (EMR) that shines inside the water facilitates momentum and kinetic energy transfer.

Having provided a definition and a classification of INTERACTIONS and INTERACTION MECHANISMS we will now reflect on the aspect of *Information*, which is commonly referred to as an interaction in literature. Following this will be some reflections on what an *interface* is and how it distinguishes itself from the other terms.

2.13.2 Capturing INFORMATION in the *Interaction Framework*

The most predominant classification of interactions in literature is *Material, Energy, and Information* (Hubka and Eder 1988; Pahl and Beitz 1988), which have been the basis for many further classifications ever since its conception (Parslov and Mortensen 2015). However the choice of having *Information* on par with *Material* and *Energy*, seem to be based primarily on a choice of convenience to adapt to the engineering design domain and may invoke ambiguity. As many authors have also pointed out (Andreasen 1980; Hubka and Eder 1988; Wie et al. 2001; Dickerson and Mavris 2010), *Information* is essentially redundant with *Material* and *Energy* transfer, e.g. flashing light (FORCE facilitating ENERGY transfer) using Morse code will facilitate transfer of information as well. Because *Information* is an essential part of engineering design today, we will attempt to clarify what *Information* is using the language of this *Interaction Framework*.

A transfer of *Information* from one system to another is synonymous with a transfer of understanding, or knowledge, regarding the state of the first system to the second system. The INTERACTION MECHANISM for facilitating this communication is not necessarily critical to the message, but the 'language' or protocol of the communication is, i.e. the sending and the receiving systems must encode and decode the *Information* according to the same protocol in order to 'get the message across'.

A well-known ISO standard called *Model of Architecture for Open Systems Interconnection (OSI-model)* standardizes the external interactions of Open Systems (Zimmermann 1980). It uses seven layers to describe the interactions in an *Information* network, with the lowest layer being a *physical layer*. A data connection may therefore be viewed from several layers of abstraction. At the base layer (i.e. physical) the data connection actually consists of a pattern of pulses with varying voltage/current levels but at a higher level, it is the instructions being sent back and forth. At an even higher level, it is the commands which are relevant, e.g. USB. So the abstraction level depends on the application, but all are based on the INTERACTION MECHANISMS; FORCE and MATERIAL transfer.

For the purpose of this framework we therefore suggest that:

- INFORMATION can only be transferred by means of INTERACTION MECHANISM; FORCE or MATERIAL transfer.
- The addition of a *common protocol* between two systems is conditional to allowing any INTERACTION MECHANISM to be interpreted as INFORMATION with meaning, e.g. the morse code protocol allows for exchange of INFORMATION (communication) between two parties using either FORCE facilitated energy transfer (e.g. sound, light etc.) or MATERIAL facilitated energy transfer (e.g. electrical signals).

2.14 WHAT IS AN INTERFACE?

On the basis of the above presented *Interaction Framework* we are now capable of defining an INTERFACE.

In this framework, an INTERFACE is perceived as an "infinitely thin" plane of separation in between two system elements. The INTERFACE does therefore not have function, meaning that there is no transformation of input to output across an interface. The INTERFACE is characterized by certain INTERFACE *conditions* that are necessary for an INTERACTION MECHANISM to take place between two physical system elements, e.g. modules or components.

An INTERFACE is part of the design solution and can as such be considered a design object. As with any other object in engineering design, an INTERFACE is matured through a synthesis process with the INTERACTION MECHANISM acting as a requirement for the INTERFACE design process.

An INTERFACE therefore changes nature from being characterized by a statement of conditions (e.g. a hole is conditional to the transfer of MATERIAL) to being a design specification of two *design features* each belonging to either module. The collective behavior of the two features constitutes the behavior of the INTERFACE which is to facilitate the INTERACTION MECHANISM, which again facilitates the INTERACTION.

It is our assessment that distinguishing between an INTERFACE, an INTERACTION MECHANISM, and an INTERACTION, will lead to more *productive designing*.

3 EVALUATION OF FRAMEWORK

In order to evaluate the effects of this framework on practice, a test was set up. The test only aims at assessing the usefulness and applicability of the framework given the limitations of the scope of the test.

The test setup was first and foremost guided by certain requirements that were instrumental in the development of the framework. See *1.4.2 Research Method*.

3.1 TEST PARTICIPANTS

In total five individual tests were carried out with various test participants (TPs). This is not sufficient to conclude based on statistical evidence. The purpose of this evaluation study is rather to provide an initial indication of the effects of the developed support. The criteria for selecting these five individual TPs were *area of technical expertise* and *years of industrial experience*. The actual details are listed below:

- Area of technical expertise
  - Primary: Mechanical, Electrical,
  - Secondary: Mechatronics, Software, Fluid Mechanics, Thermal, Hydrography, Ultrasound, Systems Engineering, software
- Years of industrial experience (Listed from TPs 1 through 5)
  - 8, 9, 12, 33, and 40 years of experience.

The reason for using these selection criteria was because of their influence on the phenomenon in question and therefore our ability to address requirements #1, #2, and #3. See table 1.

3.2 TEST METHOD REFLECTIONS

3.2.1 Test case product: Hair dryer

An early decision was made to use an existing physical hair dryer as a test case. It is most often easier for engineers to reason based on a physical object, than having to reason from an abstract description to an actual physical object. Also, by focusing on a specific interface and displaying it to the TP, we are able to measure how close their count of interactions compares to our count of interactions and use that as measure of how they progress during the test.

A downside to this approach is that we are not testing the effectiveness of the framework in a synthesis situation but rather in an analytical situation. Nonetheless, providing a user with a framework for reasoning about interactions is likely to stimulate their imagination whether it is an analytical or a synthesis exercise.

Other aspects for selecting the case product were *familiarity with the product's mode of action* as well as *level of complexity of the product.*

Because the product was disassembled and non-functioning (not connected to the power grid), it was important that all test participants were familiar with how the product worked – it's mode of action. 5 out of 5 TPs answered that they *"totally agree"* when asked to rate the following statement: "*I am familiar with how the product technically works*." It was also important to choose a product which they had not professionally worked on, since that experience would have biased their answers and made them incomparable.

The second objective was to choose a case product with a reasonable complexity with respect to the time available and the scope of the investigation, cf. table 1. A hair dryer was considered to have a fair amount of different interactions, while being rather simple (in opposition to *complicated*) to reason about. See fig. 9 for picture of test case product.

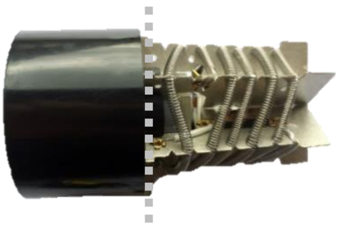


Fig. 9 The blower module (left of the interface) and the heater module (right of the interface). A physical sample was provided to the TP. The airstream flows from left to right in this image

3.2.2 Test procedure

Due to the limited number of TPs it was decided to let all participants go through the same exercise, at first without having been introduced to the framework and secondly after the introduction.

The exercise which is repeated sounded as follows:

Consider a particular boundary between two modules, see fig. 9, and:

- *List all INTERACTIONS that pass or act in that boundary (intended and unintended) given the physical sample in front of you.*
- *Specify the requirements (design input) for two of those INTERACTIONS.*
- *Specify the related characteristics of the INTERACTION MECHANISMS that the two module owners can negotiate between them.*

After having completed all three tasks, they were asked to rate three statements about how easy they found the exercise and whether they felt that they had gained a complete overview of all the interactions. Once this was completed they were introduced to the framework and asked to repeat the exercise by adding to the earlier answers. This provides us with an idea of their progression. It obviously also raises questions about whether the TPs are affected by their immediate answers and therefore cannot think outside their original mindset once asked to. Another risk was that they more or less may have captured all interactions in their first try, so when reiterating using the framework as a mental model, they already had covered most of it. These aspects of bias are mitigated through a qualitative interview at the end of the test, as well as correlated with the years of experience.

After retrying the test they are asked to rate three statements about whether the framework improved their insight into the same three questions. The results will be presented below.

3.2.3 Test results

The following chart (see fig. 10) shows how 5 out of 5 TPs identified more interactions once being introduced to the framework. On average, the TPs added 85% more interactions. While we cannot statistically conclude anything from this, it may indicate a positive effect on the identification of INTERACTIONS (Requirement #3). Further studies may be conducted to verify this indicative result.

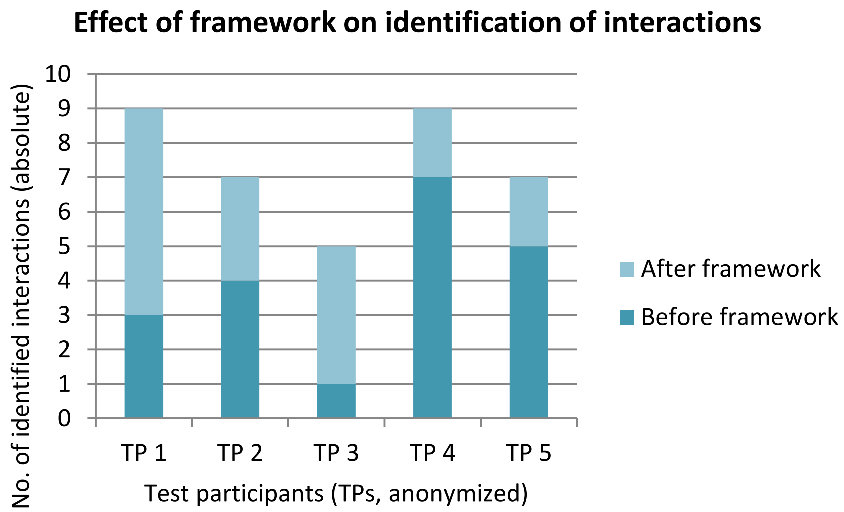


Fig. 10 This chart displays the number of identified interactions per TP, both before and after they were exposed to the framework. NB. The "after framework" is a count of the added identified interactions

A noteworthy observation is the high number of identified INTERACTIONS that the very experienced TP4 and TP5 identified prior to having been introduced to the framework compared with the other TPs. To some extent, it seems as if the framework equalizes the difference in experience levels when it comes to identifying interactions.

The following table summarizes the test results and compares them with the requirements for the framework, see table 10.

Table 10 Results of the test – includes both quantitative and qualitative observations and interview

ID#	Requirement	Conclusion	Comments
1	The **framework** shall enable the user to **identify a greater number of interactions outside his area of technical expertise** than he/she would otherwise have identified w/o the framework.	Using their intuition their observations had to do with the primary INTERACTION MECHANISM; air flow, electrical. After the framework they included more subtle INTERACTION MECHANISMS like; heat conduction, electric field force, EMR, weight of modules (grav.).	It seems as if they are more comfortable thinking outside their own area of expertise using the framework.
2	The **framework** shall allow an **inexperienced engineer** to **identify as many or more interactions** than an experienced engineer when analyzing an existing product.	The framework did seem to even out the difference in number of identified interactions despite different experience levels. TP4 and TP5 have 33 and 40 years of experience, while TP1-3 collectively has 29 years of experience.	
3	The **framework** shall enable the user to **identify more interactions** than was achieved w/o the framework	On average, the TPs added 85% more interactions once introduced to the framework.	100% of the TPs *agreed* or *totally agreed* to the statement that "the framework improved my ability to identify interactions."

Future research must investigate these effects further in a real-world setting, where a product is synthesized, e.g. one might follow a product development project and apply the framework on a single module and compare the result with other modules of similar complexity. Interesting measurable success criteria could *the number of change requests that were filed for a given module compared with one that did not follow the framework.* Another measure could be *the amount of rework (measured in engineering hours) that was used to achieve a producible module.*

4 DISCUSSION

An important driver for coming up with an alternative classification of *Interactions* and *Interfaces* has been to reduce ambiguity in the way engineers reason and communicate about interfaces and interactions in engineering design. By building the *Interaction Framework* on a physics foundation, it is possible to argue with confidence that the classification of INTERACTION MECHANISMS is both *mutually exclusive* and *complete* in terms of capturing all possible physical phenomena across any engineering discipline, given the limitations set up in the research approach.

It is therefore purposeful to discuss how this framework compares to other published contributions. We will do this by presenting two simple examples of common interactions and discuss how they are captured by the different approaches. See fig. 11.

The first interaction example considers *hot, ionized air being expelled out of a hair dryer*. The overall function of a hair dryer is to remove water/moisture from a person's hair. The most commonly applied solution principle for doing this is to blow a hot stream of air onto the hair causing the hair to whirl up and the water to vaporize. The air then consumes this moist and carries it away. Some hair dryers market themselves as *ionic* meaning that they expel charged particles (electrons) on to the user's hair in order to de-charge it and remove static electricity.

Given the purpose of the air stream there are thus certain physical properties (INTERACTIONS) of the air stream which should be mapped by the system architect in order to fully capture the solution. In the classifications posed by Pahl et al. (2007) and Hirtz et al. (2002) they do not capture the fact that a gas (i.e. material, air) carries with it physical properties such as thermal energy, charge, momentum, which are all important to the function. Too much thermal energy would burn the person, and too little would not vaporize the water. Too small a mass flow would limit the air stream's capacity to carry away moist and too high a mass flow would cause the hair to be whirled violently due to its momentum thus discomfort. In other words, these are aspects not covered by simply mapping "gas". Although "gas" may be sufficient for a very high level perspective, it does not support the system architect in reasoning about the actual physical properties that are inherently part of that gas.

In our framework, the system architect reasons top-down by asking:

- *What are the relevant INTERACTIONS that need to be transferred in order to obtain the function?*
- *Which INTERACTION MECHANISMS may facilitate this INTERACTION?*
- *What INTERFACE conditions are necessary to allow for the INTERACTION MECHANISM to transfer/act?*
- *What INTERFACE features are necessary to carry the INTERACTION MECHANSIM?*

This allows the system architect to be more solution neutral by addressing the intended function (i.e. dry hair) and not the solution (i.e. blow hot air). Also, the INTERACTION specification will be more *complete*, thus minimizing the risk of causing damage to the user or cause product failure.

Another example is the sound or acoustic energy (see dotted line in fig.11) associated with an operating hair dryer. Using the approaches posed by Pahl et al. (2007) and Hirtz et al. (2002) does not reveal the fact acoustic energy, are waves of momentum, strain energy, and kinetic energy with zero MATERIAL transfer caused by an oscillating 'contact' force. By being presented with the total list of possible INTERACTIONS related to a given INTERACTION MECHANISM, the system architect is able to assess the relevance of a given INTERACTION, e.g. the momentum and kinetic energy of a pressure wave following an explosion can cause great damage to its surrounding.

In general, it can be observed that the current classifications seem primarily concerned with identifying the *intended* INTERACTIONS and therefore may miss the *unintended* INTERACTIONS that may arise due to the chosen INTERACTION MECHANISM. With our framework, the system architect is confronted with all of the possible INTERACTIONS that are associated with a given INTERACTION MECHANISM whether intended or not.

This has some advantages to our treatment of INFORMATION. Whilst INFORMATION is just a common protocol using this *Interaction framework* means that INFORMATION cannot exist without the underlying INTERACTION MECHANISM, and therefore the *unintended* INTERACTIONS also get "exposed" in addition to those that are used for the INFORMATION transmission. For example, if you transmit INFORMATION with sound, this framework makes the designer aware of the force applied by the INFORMATION transmission (sound pressure levels). So by coupling the INFORMATION to physical INTERACTIONS, the unintended INTERACTIONS that could arise will not be forgotten.

A consequence of this more nuanced approach to documenting INTERACTIONS is that the amount of data related to each relation in a system, will increase drastically. The reality is however, that the physical properties are there, whether or not it is documented. With this framework, the system architect is able to assess the relevance of a given INTERACTION MECHANISM or INTERACTION and omit it if found irrelevant at the time of reflection.

With the increased attention to MBSE, more data will not necessarily become an issue because of sophisticated software tools. It is rather the *right data* which become critical to ensuring successful integration, which this framework attempt to support.

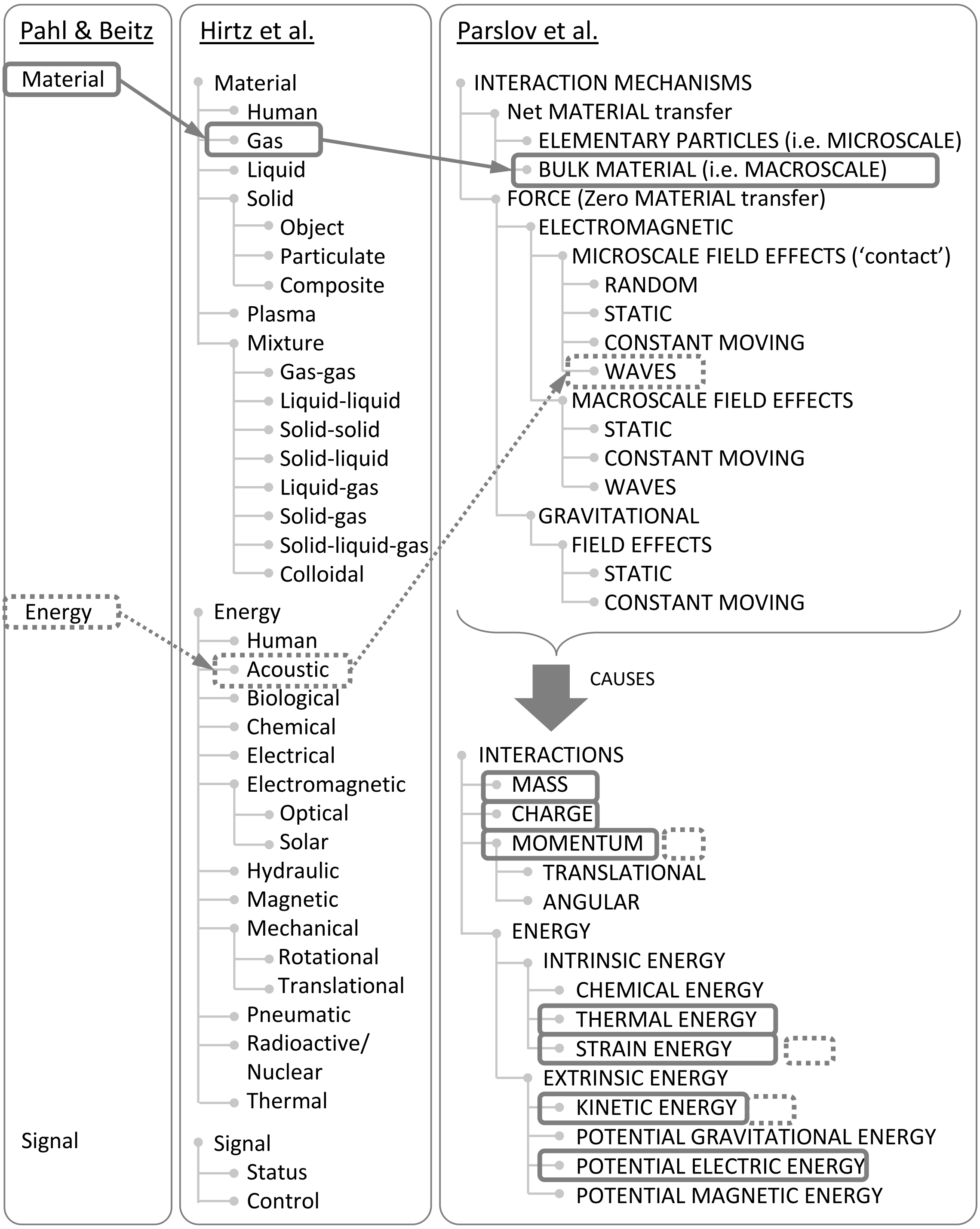


Fig. 11 This figure compares proposed framework with other approaches using examples of hot air and sound from a hair dryer. The solid line shows how hot, ionized air from a hair dryer is captured using all three approaches. The dotted line exemplifies sound

5 CONCLUSION

This paper presents a framework for understanding interactions and interfaces in the engineering design domain. The aim is to reduce ambiguity during the architectural decomposition of complex multi-disciplinary systems by creating a common language of interfaces and interactions across any technical disciplines deduced from fundamental physics. At the core of the framework is a distinction and classification of INTERACTION MECHANISMS, INTERACTIONS,

and INTERFACES which serve as a useful mental model for reasoning about system relations when designing while being compliant with the laws of physics.
The framework is evaluated in five expert user tests, which indicate a positive effect of the framework in terms of enabling the users to identify more interactions outside their own area of technical expertise. Also, the framework shows a positive effect in terms of supporting users with less experience to identify as many interactions as more experienced users. While there is not enough data to conclude with statistical confidence future research might look at how to apply the framework in practice and perform a case study in a real-world project.
If we envision a future where complex, multi-disciplinary products are designed and produced from an end-to-end model based simulation environment we need a consistent and rigorous theoretical foundation for understanding and defining interactions and interfaces. It is the ambition of this *Interaction framework* to contribute to this vision.

6 REFERENCES

Albers A, Wintergerst E (2014) The contact and channel approach (C&C2-A): relating a system's physical structure to its functionality. In: An Anthology of Theories and Models of Design. Springer, pp 151–171

Andreasen MM (1980) Machine design methods based on a systemic approach. Lund University

Andreasen MM (2011) 45 Years with design methodology. J Eng Des 22:293–332.

Andreasen MM, Hansen CT, Cash P (2015) Conceptual Design: Interpretations, Mindset and Models, 1st edn. Springer, Kgs. Lyngby

Andreasen MM, Howard TJ, Bruun HPL (2014) Domain Theory, Its Models and Concepts. In: An Anthology of Theories and Models of Design. pp 173–196

Bettig B, Gershenson J (2010) The representation of module interfaces. Int J Prod … 10:291–317.

Blees C, Krause D (2008) On the development of modular product structures: a differentiated approach. In: International Design Conference. Dubrovnik, pp 301–308

Blyler J (2004) Interface Management - Managing complexity at the system interface. Instrum. Meas. Mag. IEEE 7:32–37.

Buede DM (2012) The Engineering Design of Systems: Models and Methods, 2nd edn. Wiley

Chabay RW, Sherwood BA (2011) Matter and Interactions, 3rd edn. Wiley

Crawley E, Cameron B, Selva D (2015) Systems Architecture: Strategy and Product Development for Complex Systems. Prentice Hall

Crawley E, Weck O de, Eppinger S (2004) The influence of architecture in engineering systems. MIT esd

Dickerson CE, Mavris DN (2010) Architecture and Principles of Systems Engineering. CRC Press

ECSS (2015) Space Engineering - Interface Management. 1–54.

Grady JO (1994) System integration. CRC press

Haskins C, Forsberg K, Krueger M (2006) Systems Engineering Handbook. INCOSE

Hirtz J, Stone R, McAdams D (2002) A functional basis for engineering design: reconciling and evolving previous efforts. Res Eng … 13:65–82. doi: 10.1007/s00163-001-0008-3

Hubka V, Eder WE (1988) Theory of Technical Systems, 2nd edn. Springer-Verlag, New York, NY

Kapurch SJ (2007) NASA Systems Engineering Handbook. NASA Spec Publ 360.

Krause D, Eilmus S, Jonas H (2013) Developing Modular Product Families with Perspectives for the Product Program. In: Smart Product Engineering. Springer Verlag Berlin Heidenberg, pp 543–552

Lalli V, Kastner R, Hartt H (1997) Training Manual for Elements of Interface Definition and Control. National Aeronautics and Space Administration, Office of Management, Scientific and Technical Information Program

Liang V-C, Paredis CJJ (2004) A port ontology for conceptual design of systems. J Comput Inf Sci Eng 4:206–217. doi:

10.1115/1.1778191

Liedholm U (1999) Analysis of purposeful and incidental interactions. In: ASME Design Engineering Technical Conference Proceedings, DETC99/DTM.

Malmqvist J (1993) Towards Computational Design Methofs for Conceptual and Parametric Design. Chalmers University of Technology

Pahl G, Beitz W (1988) Engineering Design: A Systematic Approach, 1st edn. Springer

Pahl G, Beitz W, Feldhusen J, Grote K-H (2007) Engineering Design: A Systematic Approach, 3rd edn. Springer

Parslov JF, Mortensen NH (2015) Interface definitions in literature: A reality check. Concurr Eng. doi: 10.1177/1063293X15580136

Pimmler TU, Eppinger SD (1994) Integration analysis of product decompositions. In: ASME Design Theory and Methodology Conference Minneapolis, MN. Alfred P. Sloan School of Management, Massachusetts Institute of Technology,

Rahmani K, Thomson V (2012) Ontology based interface design and control methodology for collaborative product development. Comput Des 44:432–444. doi: 10.1016/j.cad.2011.12.002

Rodenacker W (1971) Methodisches konstruieren. Springer, Berlin

Stone RB, Wood KL (2000) Development of a functional basis for design. J Mech Des 122:359.

Szykman S, Racz JW, Sriram RD (1999) The representation of function in computer-based design. In: Proceedings of the 1999 ASME design engineering technical conferences (11th international conference on design theory and methodology).

Uddin A, Felician C, Mohammed KK (2015) Development of an Interface Analysis Template for System Design Analysis. In: DS-80-4 Proceedings of the 20th International Conference on Engineering Design (ICED 15) Vol 4: Design for X, Design to X. Milan, Italy,

Ulrich KT, Eppinger SD (2012) Product design and development. McGraw-Hill Higher Education, New York

Weck OL De, Roos D, Magee C (2011) Engineering systems: Meeting human needs in a complex technological world. MIT Press

Wheatcraft LLS (2010) Everything you wanted to know about interfaces, but were afraid to ask. Compliance Automation

Wie MJ Van, Greer JL, Campbell MI, et al (2001) Interfaces and Product Architecture. In: ASME 2001 International Design Engineering Technical Conferences And Computers and Information in Engineering Conference. pp 1–12

Zheng C, Duigou J Le, Bricogne M, Eynard B (2016) Multidisciplinary Interface Model for Design of Mechatronic Systems. Comput Ind 76:24–37.

Zimmermann H (1980) OSI Reference Model--The ISO Model of Architecture for Open Systems Interconnection. IEEE Trans. Commun. 28:

Defining Interactions and Interfaces in Complex Multi-Technological Products – A Multi-disciplinary, Physics-based Approach

Jakob Filippson Parslov[1], Barney Gerrard[1], and Niels Henrik Mortensen[2]

[1]New Product Development, R&D
Radiometer Medical ApS
Åkandevej 21
2700 Brønshøj, Denmark
Email: jakobparslov@gmail.com
& barney.gerrard@gerrard.dk
Tel.: +45 60152745
Corresponding author: Jakob F. Parslov

[2]Section for Engineering Design and Product Development
Department of Mechanical Engineering
Technical University of Denmark
Produktionstorvet, Building 426
2800 Kgs. Lyngby, Denmark
Email: nhmo@mek.dtu.dk

ACKNOWLEDGEMENT

We would like to express our gratitude to Radiometer Medical ApS for co-financing this research together with the Innovation Fund Denmark as part of J.F.P's doctoral studies.

ABSTRACT

Products are becoming increasingly multi-technological and complex because companies strive for superior product functionality and performance in order to compete on a global market place. This means that product development becomes increasingly multi-disciplinary and therefore 'multi-lingual' thus increasing the risk of miscommunication and ultimately rework due to incompatibilities at the product interfaces. This paper presents a new approach to defining *interactions* and *interfaces* in complex multi-technological systems aiming at reducing ambiguity and increasing completeness of interaction and interface descriptions. The multi-disciplinary approach is based on a physics-based *Interaction framework*, which is first summarized and then further extended through an elaboration of requirements and specification as well as an in-depth treatment of the nature of an interface. This is followed by an 8-step architecting approach describing how to use the extended framework in practice and a tool for supporting the specification of interactions. The *Interaction and Interface framework* and the tool have been tested, which indicate a positive effect on the test participants' ability to specify interactions consistently and unambiguously. Future research must verify this effect in a real-life case study in order to prove the usefulness of the *Interaction and Interface framework*.

1 INTRODUCTION

1.1 PROBLEM STATEMENT

Products and product development are becoming increasingly multi-technological and complex due the increasing functionality, optimized performance and advancement of technology, which necessitate the use of multiple engineering disciplines in the development of a product (Fotso and Rettberg 2012). One of the challenges with this diversity of engineers is a lack of common language and common mental models (Jarratt et al. 2004).

According to research, many problems, if not most problems, occur at the interfaces in a system during development (Grady 1994; Kapurch 2007; Wheatcraft 2010; Buede 2012). Faced with a lack of a common language to speak about interactions and interfaces, engineers might use abstract language to communicate, which introduces the risk of misinterpretation and ultimately rework because of incompatibilities at the interfaces (Parslov and Mortensen 2015).

Also, the classifications of interactions and interfaces that exist today lack a few very important characteristics; they are not mutually exclusive, they are not complete in terms of covering all technical disciplines, and finally they do not prescribe or guide the user in reasoning clearly from a higher abstraction level down to the embodiment phase (Zheng et al. 2016). This decomposition of the architecture is a critical part of a system architect's role of consistently ensuring system functionality and performance (Albers and Wintergerst 2014; Bonnema et al. 2015).

As a result of the above there is a need for a theoretical framework that supports systems architects in reasoning about interactions and interfaces rigorously and consistently through the early development process.

This paper describes the broader, operational part of a theoretical *Interaction framework*, which was thoroughly described in Parslov et al. (2016). For that reason we refer to the associated paper for an overview of related work and for more details on how the *Interaction framework* was derived (Parslov et al. 2016).
The research questions which are investigated in this paper are:

1. *How can an INTERFACE be defined and characterized, based on the understanding from the Interaction framework?*
2. *How can the extended Interaction and Interface framework and tool be applied in practice to support complete and unambiguous INTERACTION and INTERFACE specifications?*

We do therefore not constrain ourselves to the early architectural phase, but reflect upon the use of the framework from early architectural decomposition to the embodiment of module interfaces.

The paper is structured as follows; firstly we introduce a summary of the *Interaction framework* which supports multi-disciplinary interaction reasoning. We then present an extension to the framework with a prescription of how to describe interaction requirements and specifications followed by a definition and classification of an *interface* and its relation to the *Interaction framework*. We then introduce an 8-step architecting approach for applying the framework in practice accompanied by a tool called an *Interaction Specification Wheel (ISW).* The framework in combination with the tool is then evaluated and finally the results will be discussed and concluded on.

1.2 RESEARCH APPROACH

This paper represents the result of a 2.5 year research effort into the nature of interfaces and interactions. The research is characterized by a first principles approach, in the sense that the classification that is summarized below as part of the framework is derived from fundamental physics with respect for the phenomena of engineering design.

This paper features four contributions:

- A characterization of INTERACTION requirements and specifications
- A definition and classification of an INTERFACE based on the *Interaction framework*
- An 8-step architecting approach for applying the *Interaction framework.*
- An *Interaction Specification Wheel* (ISW) for operationalizing the framework

The two first contributions are extensions to the *Interaction framework* and build on an understanding of the phenomena of engineering design. The two latter contributions prescribe how to apply the framework in practice. The purpose of the concepts is to improve work practices of system architects by providing *useful* means of support. The framework and tool targets senior engineers, system architects, or system engineers at companies developing multi-technological, complex products, where the risk or cost of failure is high. Being systematic about documenting the architectural phase of product development is an investment, which will earn itself in through fewer integration issues, less rework, and shorter time-to-market. Even for companies developing less complex products, adopting this way of thinking about interactions and interfaces might be of great benefit in terms of improving communication and collaboration.

The theoretical background for this paper is systems theory, in particular Theory of Technical Systems (Hubka and Eder 1988), which provide a foundation for speaking about products as technical systems, which is fully compatible with the *Interaction framework*.

1.2.1 Research method

The following requirement was set up to guide the further development of the framework and the tool. See Table 1.

Table 1 This table shows the requirement, the influencing factor, how we measured the results, the related research question and comments.

Requirement	Influencing factor	Measurable criteria	Research Question	Comments
The **framework and tool** shall lead to a **less ambiguous specification** of the interactions across different technical disciplines, than could be achieved without the support.	Technical background & experience	Number of different types of attributes/ properties (consistency)	RQ 2	Support for reducing ambiguity in interaction specifications

The purpose of this paper is therefore to extend the *Interaction framework*, in order to reduce ambiguity in the specification of INTERACTION. The *Interaction framework* and the tool have in part been developed up against this requirement. For other requirements regarding the framework we refer to (Parslov et al. 2016).

2 SUMMARY OF THE INTERACTION FRAMEWORK

The aim of the *Interaction framework* is to equip system architects with a mental model and mindset for reasoning about *interaction* across any engineering discipline, at any level of scale as part of the architectural decomposition in early phase product development. The framework was derived using a first principle, physics-based approach in order to arrive at a mutually exclusive and collectively exhaustive classification tailored for engineering design.

Any physical SYSTEM possesses *conserved properties* such as *translational (T) momentum*, *angular (A) momentum*, and *energy*. A product can be considered as a system, named a technical system (Hubka and Eder 1988) with the same conserved properties; Momentum (T&A) and energy. The sum of these properties for any given technical system constitutes the state of the technical system.

The only way of changing the state of a system, is by infusing or extracting momentum (T&A) and energy across the system boundary. We call this transfer of physical properties for INTERACTIONS.

While INTERACTIONS are instrumental to state changes, they do not happen without a cause. They must be *facilitated* by some physical phenomenon, which we denote INTERACTION MECHANISMS.

There are two kinds of INTERACTION MECHANISMS; Force and MATERIAL transfer. Thus, whenever a force is present at the system boundary (with zero MATERIAL transfer), or MATERIAL is transferred across the system boundary, various kinds of INTERACTIONS might occur. In other words, INTERACTION MECHANISMS are the *causes* of INTERACTIONS, representing the *effect.*

The rows in the following table illustrate the systematic classification of INTERACTION MECHANISM. The *primary classification* has to do with the fact that the abstraction level of engineering design is much higher than physics. The *secondary classification* separates the two fundamental forces of nature, which are accountable for most physical phenomena in engineering design. The *tertiary classification* has to do with the aspect of length scale and the *fourth classification* considers the behavior, or pattern of movement of the INTERACTION MECHANISM.

Table 2 Classification of INTERACTION MECHANISMS and INTERACTIONS and how they relate (Parslov et al. 2016)

						INTERACTION (EFFECT)	
	PRIMARY (ABSTRACTION)	SECONDARY (TYPE)	TERTIARY (LENGTH SCALE)	QUATERNARY (PATTERN OF MOVEMENT)	EXAMPLES USING FAMILIAR DOMAINS	**TRANSFER OF MOMENTUM** - (TRANSLATIONAL & ANGULAR)	**TRANSFER OF ENERGY**
INTERACTION MECHANISM (CAUSE)	**FORCE**	ELECTRO-MAGNETIC	MICROSCALE FIELD EFFECTS (PHYSICAL CONTACT)	RANDOM	*Thermal conductivity, stove, radiator etc.*	*NO*	THERMAL
				STATIC	*Assembly interfaces*	*NO*	*NO*
				CONSTANT MOVING	*Crane lifting container, compression of material, rotating shaft etc.*	YES	STRAIN
				WAVES	*Pistons, sound, earthquakes etc.*	YES	STRAIN, KINETIC
			MACROSCALE FIELD EFFECTS	STATIC	*Balloon on a jumper, permanent magnet/ electromagnet*	*NO*	*NO*
				CONSTANT MOVING	*Solenoid (constant current increase assumed)*	YES	ELECTRIC POT., MAGNETIC POT.
				WAVES	*EMR (i.e. sunlight, x-rays, UV-light, induction etc.)*	YES	KINETIC, ELECTRIC POT., MAGNETIC POT.
		GRAVITATIONAL	-	STATIC	*Earth's field (approx.)*	*NO*	*NO*
				CONSTANT MOVING	*Black holes with constant mass gain*	YES	GRAVITATIONAL POT.
	MATERIAL TRANSFER	-	ELEMENTARY PARTICLES (MICROSCALE)	CONSTANT MOVING	*Electricity, electrolysis, osmosis, diffusion etc.*	YES	KINETIC, GRAVITATIONAL POT., ELECTRIC POT., MAGNETIC POT.
			BULK MATERIAL (MACROSCALE)	CONSTANT MOVING	*Hydraulics, pneumatics, advection, etc.*	YES	CHEMICAL, THERMAL, STRAIN, KINETIC, GRAVITATIONAL POT., ELECTRIC POT., MAGNETIC POT.

These classes thus covers all physical phenomena relevant to engineering design, whether viewing it from an electrical, mechanical, thermal etc. point of view. Transfer of *Information* from one system to another is synonymous with the transfer of understanding or knowledge about the state of the sending system. Thus, *Information* can only be transferred by means of an INTERACTION MECHANISM as well as the addition of a common protocol prescribing how to interpret the INTERACTION MECHANISM as *Information*, e.g. the morse code protocol allows for flashing lights (INTERACTION MECHANISM) to be translated into meaningful *Information*.

On the right hand side of Table 2 is a high level classification of INTERACTION into momentum (T&A) and energy. Energy can further be classified into:

- Intrinsic energy
    - Chemical energy
    - Thermal energy
    - Strain energy

- Extrinsic energy
    - Kinetic energy
    - Potential gravitational energy
    - Potential electric energy
    - Potential magnetic energy

The difference between *intrinsic* and *extrinsic energy* is the fact that *intrinsic energy* is an inherent property of the MATERIAL independent of its surroundings, whereas the *extrinsic energy* requires an external Frame of Reference (FoR) to make sense. For MATERIAL transfer, it might be useful to also include *mass transfer* and *charge transfer* even though they can be described using the above stated energies.

The intersection between the INTERACTION MECHANISM (rows) and INTERACTION (columns) lists the possible INTERACTIONS relative to any given INTERACTION MECHANISM. The powerful thing about this framework, in comparison with other classifications, is that it is *mutually exclusive* and *collectively exhaustive* from a physics standpoint. It means that any physical phenomenon, indifferent from technical discipline, may only fall into *one* of the eleven detailed classes of INTERACTION MECHANISMS.

Due to the complex behavior of the physical world, a system architect may choose to *superimpose* the INTERACTION MECHANISMS in order to fully describe a certain physical phenomenon in a complex product. This principle of superimposition therefore enables the system architect to reason more freely about any physical relation across all technical disciplines.

Product design seldom starts purely from a top-down, *synthesis* perspective although this may be the theoretically ideal. Often, development teams for example use working assumptions, jumps to solutions before considering the functional aspect, reuse existing solutions, or buy off-the-self components, which forces them to *analyze* the impact of their decision, possibly do rework, and continue synthesizing the solution.

The *Interaction framework* supports this phenomenon by allowing for *forward* and *backward* reasoning between INTERACTION and INTERACTION MECHANISM using Table 2 as preferred. See Fig. 1.

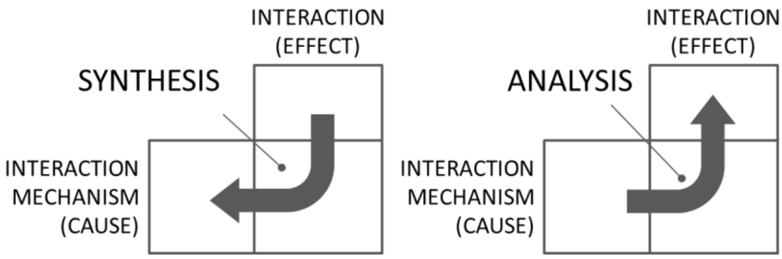


Fig. 1 Depiction of two reasoning patterns (left: synthesis, right: analysis) from intended INTERACTION to applicable INTERACTION MECHANISMS and vice versa. The matrix in the background depicts Table 2

3 EXTENDING THE FRAMEWORK

The *Interaction Framework* has so far introduced two new concepts; INTERACTION and INTERACTION MECHANISM. However, in order for these terms to be applicable in practice, they must comply with some of the common practices of engineering design. The following section will therefore introduce a general model of design processes and apply this thinking for characterizing the concept of INTERACTION requirements and specifications.

Lastly we will extend the framework with an in-depth treatment of INTERFACE in relation to the *Interaction Framework*.

3.1 GENERAL DESIGN PROCESS MODEL

Design control is a widely used practice in new product development, especially in highly regulated branches such as the medical device branch (Liu 2013). Design control is primarily about sound *design processes* a in order to ensure that devices or products meet user needs, intended uses and specific requirement in safe and non-harmful way. Although design control might not be relevant for all companies, the *systematic thinking* behind design control is valuable to any company doing product development and is very much aligned with thinking in systems engineering (Haskins et al. 2006; Kapurch 2007).

The following illustrates a general model of *designing*. See fig. 1.

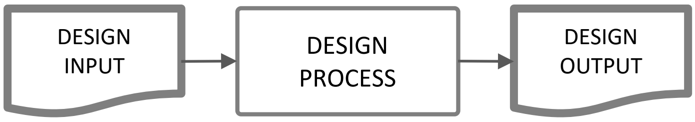


Fig. 2 General model of *designing.* A *design process* is guided by a *design input* and produces a *design output*. A squared box is a process, and a whirled box is a document

In general, any *design process* is guided by an input of some kind and produces an output. The former is called a *design input* and guides the design process by setting up constraints. During the design process, the designer synthesizes solutions that satisfy the *design input* constraints, and produces a *design output*. *Verification* of the design process involves comparing whether the design output complies with the design input. It is often the case that the *design output* from one *design process* becomes *design input* to another *design process*. Any object, which needs *designing* is subject to this model. This includes not only products as a whole but also INTERACTIONS and INTERFACES as parts of a product.

In many companies, it is common to consider a *design input* as a *requirement* and a *design output* as a *specification*. However, our concern with this is that sometimes, the purpose of a design process is to create *requirements* meaning that the design output would become *requirements specifications*. In order to avoid any confusion we simply rely on the terms presented in fig. 1. We will therefore use fig. 1 as a mental model as we proceed with describing the extension of the framework.

3.2 DEFINING INTERACTIONS – REQUIREMENTS AND SPECIFICATIONS

Applying the *Interaction framework* in practice involves three high-level design processes, see Fig. 3. Each design process has a design input and a design output, which vary depending on the process.

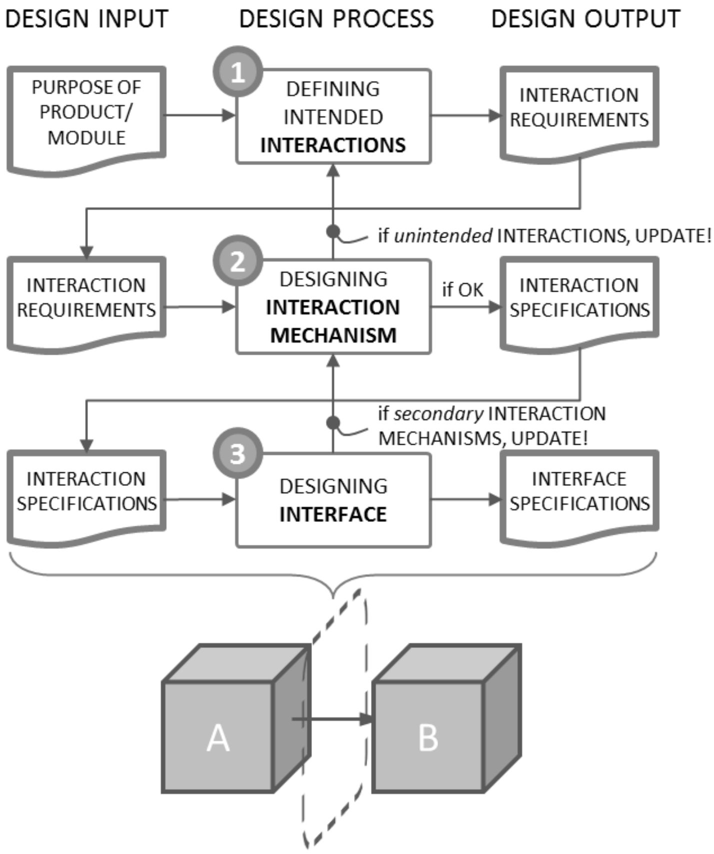


Fig. 3 Illustration of the high-level INTERACTION and INTERFACE design process, starting at step 1. The arrows merely show the reading direction

The following section will concentrate on characterizing INTERACTION requirements and INTERACTION specifications.

3.2.1 INTERACTION requirements

Designing is characterized by being highly iterative with a constant shift of mindset from *synthesizing* solutions based on the overall function of the product to *analyzing* whether the invented solutions complies with the overall function. In product development, the *behavioral attributes* (i.e. function and performance levels) are documented using *requirements*, which are intended as, standardized, measurable and unambiguous textual statements. Thus in principle, requirements should be solution neutral and describe *what* the product or module should do and not *how*. In practice however, this is seldom the case (Wheatcraft 2010).

In continuation of the above, INTERACTION requirements, are verifiable and unambiguous, textual statements of *intended* and *unintended* INTERACTIONS across a system boundary. Companies typically operate with two kinds of requirements documents; System level requirements and subsystem (low-level/module) requirements. INTERACTION requirements are typically documented in system level requirements documents, thus governing the external and internal functional INTERACTIONS between the product and its environment and between the subsystems respectively.

INTERACTION requirements capture the *intended* INTERACTIONS based on an understanding of the overall purpose or function of the product is, see Fig. 3, step 1. The *Interaction framework* provides a list of possible INTERACTIONS. Their values are typically defined using a range, in order to leave room for the subsequent design process, variations in production, or to allow for multiple modes of state (Liu et al. 2015). *Unintended* INTERACTIONS will arise as a result of the synthesis process, because choosing an INTERACTION MECHANISM may facilitate INTERACTIONS, which are not required and therefore unintended. These unintended INTERACTIONS must be accounted for in the INTERACTION requirements document as well.

3.2.2 INTERACTION specifications

Whereas the INTERACTION requirements are completely solution-neutral, they need some physical solution principle to facilitate their transfer, i.e. INTERACTION MECHANISM. The intended INTERACTIONS from the INTERACTION requirements document therefore act as input for designing the INTERACTION MECHANISM, see Fig. 3, step 2.

INTERACTION specifications are verifiable and unambiguous, textual statements of the *actual, realized INTERACTIONS*. In other words, an INTERACTION specification contains a description of the actual transfer of INTERACTION which should be in compliance with the requirements.

An INTERACTION specification contains a list of equations linking characteristics about the chosen INTERACTION MECHANISM to the intended INTERACTION. Values are assigned to each of the characteristics by selecting an appropriate INTERACTION MECHANISM, e.g. if bulk MATERIAL transfer is chosen as INTERACTION MECHANISM, a number of equations related to each of the facilitated INTERACTIONS can be set up. An example of an equation for bulk MATERIAL transfer is thermal energy transfer rate:

Thermal energy transfer rate = <u>area of the flow</u> X <u>velocity of the flow</u> X <u>volumetric heat capacity</u> X <u>temperature</u>

Multiplying the underlined characteristics of the INTERACTION MECHANISM will equal the INTERACTION. See the *INTERACTION specification template* in Appendix A, Table 6-7 for a complete overview of these equations, including SI units and dimensions for consistency checking (Mahajan 2014). Once all of the equations have been set up and values defined, these are documented an INTERACTION specification document which become a design input to the INTERFACE design process, see fig. 2 step 3.

In order to address the design of INTERFACES we will now characterize what an INTERFACE is in relation to the *Interaction framework*.

3.3 DEFINITION OF AN INTERFACE

A system boundary defines ‘what is inside the system’ from ‘what is outside the system’. The place where INTERACTIONS and INTERACTION MECHANISMS act or crosses the system boundary is called an INTERFACE.

3.3.1 What characterizes an INTERFACE?

An INTERFACE is a set of *conditions* that need to be fulfilled in order for the INTERACTION MECHANISM to take place. At an early, immature stage, these INTERFACE conditions constitute basic properties of the systems such as *permeability*, *absorbance*, or *openness* for MATERIAL transfer or *impermeability*, *resistance*, or *closedness* for FORCE. See Table 3.

Table 3 List of INTERFACE conditions derived from the choice of INTERACTION MECHANISM.

INTERACTION MECHANISM	**INTERFACE conditions**
EM* MICROSCALE FIELD EFFECTS	Impermeability to matter, resistance
EM* MACROSCALE FIELD EFFECTS	Permeability to electromagnetic field force
GRAVITATIONAL FIELD EFFECTS	N/A**
ELEMENTARY PARTICLES (MICROSCALE MATERIAL transfer)	Permeability to elementary particles, conductivity
BULK MATERIAL (MACROSCALE MATERIAL transfer)	Permeability to bulk material, openness or absorbance

** Electromagnetism. **The gravitational field force, is unaffected by any physical objects, in the sense that it is not distorted or impeded by having to “pass” through an object. It does therefore not inform any INTERFACE conditions.*

As can be seen from Fig. 3, step 3, the choice of INTERACTION MECHANISM has an influence on the INTERFACE design, in the sense that the INTERFACE conditions are determined by whatever mechanism is chosen, e.g. in order for an INTERFACE to allow the transmission of strain energy through an applied FORCE between a brake pedal and a person's foot, the two must be rigid enough so that one will not collapse or break when INTERACTED with. One could argue that the stiffness or lack hereof, of either the brake pedal or the person's foot is a characteristic of the system elements' respectively and not the INTERFACE itself. While this is a valid observation, it may however still be relevant to capture in the INTERACTION specification the fact that they need to be compatible at the INTERFACE without defining the interfacing systems themselves. INTERFACE conditions are thus design input to the INTERFACE design process.

At a later and more developed stage, the INTERFACE is embodied into a set of *physical design features* each belonging to each interfacing module. These INTERFACE features are the physical objects that facilitate the INTERACTION MECHANISM. There is likely to be a mutual dependency between the two INTERFACE features meaning that a change to an INTERFACE feature in one system requires a change to an INTERFACE feature in the counterpart system too.

The INTERFACE is therefore essentially part of the solution space. It has purpose of being (i.e. transmit force, transmit material), but does not have function (i.e. no transformation of input to some other output). The INTERFACE features are characterized by form, dimensions, shape, material, surface quality. Multiple INTERACTIONS and INTERACTION MECHANISMS are valid for a given INTERFACE.

3.3.2 How to perceive and model an INTERFACE?

As shown in Parslov and Mortensen (2015) the manifestation of an INTERFACE may be perceived very differently. This section will therefore propose a mental model of an INTERFACE which is in compliance with the *Interaction Framework*.

The common characteristic of the various instances of the mental model is that an INTERFACE is considered to be an 'infinitely thin' concept that separates one system from another and, which applies to both functional and physical modeling viewpoints, as well as on any level of abstraction. In this way, the model enforces a distinct difference between a system element, like a component or module, and an INTERFACE, i.e. an electric cable cannot be an INTERFACE, but is a component. See Fig. 4.

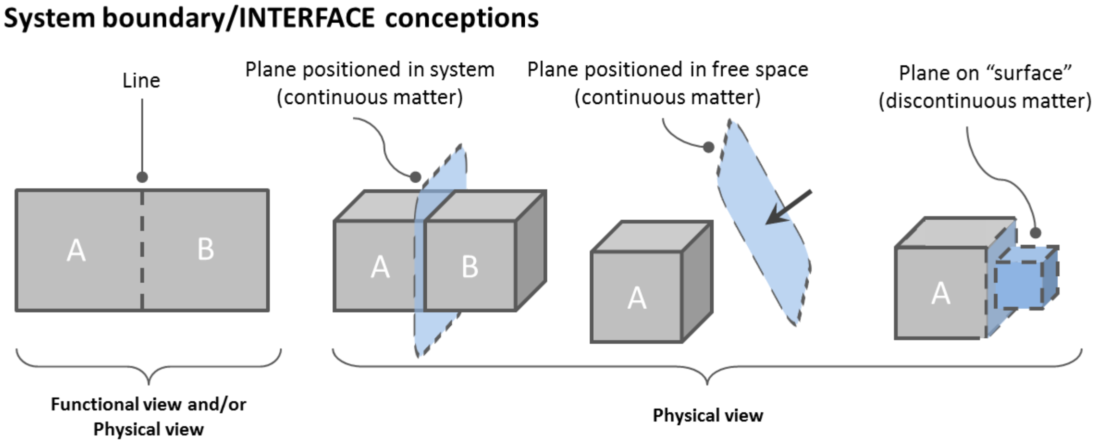


Fig. 4 A system boundary may be perceived differently depending on the modeling viewpoint. Common to all is the notion of "infinitely thin" system boundary with zero function

A technical system can be viewed from both a functional and a physical perspective (Hubka and Eder 1988). When modeling the system from a functional point of view, the system boundary may be considered as *a line* with zero physical manifestation. It merely serves the purpose of dividing the system into subsystems.

When modeling the system from a physical viewpoint the conception is *a plane*. The INTERFACE plane is considered to be "infinitely thin" and is not shared between the systems; it is not part of either one, but is conceptually positioned between them, e.g. an INTERFACE on the surface of window glass does neither consist of glass nor of air but is considered to be in between them. See fig.4.

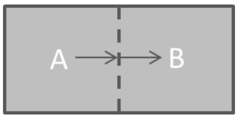
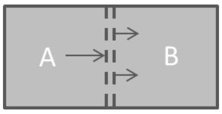
As can be seen from fig. 4, the notion of an ’infinitely thin’ plane can be positioned anywhere in and around a system. We distinguish between *continuous* matter and *discontinuous* matter. The *continuity* refers to the phase of the matter, i.e. solid, gas, liquid, plasma, and whether the placement of the INTERFACE plane is positioned at the point of phase change (discontinuous matter) or in single-phase matter (continuous matter). As a rule of thumb, any “surface” indicates a placement at *discontinuous matter* - anywhere else indicates a placement in *continuous matter*. The notion of *continuity of matter* serves the purpose of providing a term to address certain physical phenomena at INTERFACES like e.g. frictions or refraction.

3.3.3 Classification of INTERFACES

In some special cases transformation, and therefore , function arise at an INTERFACE as a result of the chosen physical solution principle for the design of an INTERFACE, e.g. a brake pad system generates heat at the INTERFACE due to forced molecular deformations (i.e. friction) or spectacles refract light as it passes from air to glass to air. Zooming in at an atomic scale would probably reveal a gradual transformation due to various interatomic INTERACTIONS thus allowing us to place an INTERFACE where the rule of *zero function* still holds. However from a product scale perspective this is not very practical, because system architects and designers simply do not reason at that level of scale.

In order to cope with these common physical phenomena in this *Interaction framework*, where INTERACTIONS and INTERACTION MECHANISMS are described at ‘infinitely thin’ interfaces, we must introduce the notion of *abstraction* to the INTERFACE concept. This is merely done out of convenience to the design process and does not violate the earlier characterizations. Table 3 show two types of INTERFACES; simple and complex.

Table 4 Classification of two types of INTERFACES and their key characteristics

GENERAL CHARACTERISTICS	
SIMPLE INTERFACE (SI)  	COMPLEX INTERFACE (CI) 
The SI is ‘infinitely thin’, i.e. has zero thickness, because otherwise it would have form and therefore function.	The CI is ‘thin’, but not ‘infinitely thin’ because the physical effects that give it function need physical length to occur.
The SI can either be defined at a discontinuity of materials (i.e. a physical surface) or at an arbitrary position in a continuum of material or space.	The CI must be defined at a discontinuity of materials (i.e. a physical surface) – without the discontinuity there cannot be function.
If the SI is defined at a physical surface, then it can react forces and moments, and deflect.	The CI can react forces and deflect. It can also accept the transfer of material through it.
	The two systems may move relative to each other, but only in the surface/plane of the CI.
CHARACTERISTICS NEEDED TO ENSURE CONSERVATION OF FORCE, MOMENTUM AND ENERGY	
There must be a force/moment equilibrium at the SI	There must be a force/moment equilibrium at the CI
MATERIAL passing through the SI must not transform or change STATE in any way – i.e. the MOMENTUM and ENERGY leaving one system is identical to that entering the other.	*Momentum* and *energy* may be divided, combined or transformed (this equals function), provided that CONSERVATION of MOMENTUM and ENERGY is respected.

	Total *momentum* (vector sum) entering and leaving the CI must be equal.
	Total *energy* (scalar sum) leaving one system equals the *energy* entering the other (zero *energy* loss or gain in the CI)

Complex interfaces should as such attract more attention during architecting, because of their functional nature, which will add further constraints on the INTERFACE design.

3.3.4 INTERFACE design triggers secondary INTERACTION MECHANISM

A noteworthy observation here is the fact that depending on the chosen INTERFACE solution, e.g. bolted mechanical INTERFACE instead of glued may result in different *local* FORCES being transmitted, even though the total FORCE transmitted by the INTERFACE is the same. We denote these local FORCES as secondary INTERACTION MECHANISMS that arise as a result of the embodiment of the INTERFACE.

In an engineering design context we therefore speak about the following types of INTERACTIONS and INTERACTOIN MECHANISMS:

- *Intended* INTERACTIONS: Those that serves the purpose of the system
- *Unintended* INTERACTIONS: Those that are inherently facilitated by the chosen INTERACTION MECHANISM but does not serve the purpose of the system
- *Primary* INTERACTION MECHANISMS: Those that are the primary mean of facilitating the intended and unintended INTERACTIONS
- *Secondary* INTERACTION MECHANISMS: Those that emerge from the embodiment of an INTERFACE and thus supports the INTERFACE in transferring the primary INTERACTION MECHANISM.

These distinctions will be used throughout the following 8-step architecting approach.

4 HOW THE FRAMEWORK SHOULD BE USED DURING ARCHITECTING

This section prescribes an 8-step, top-down architecting approach to support the application of the *Interaction and Interface framework* in practice. It is explained using a simple example of a *hair drying device*. The purpose of this architecting approach is not to contribute to the research area on decomposition as such. Rather, the 8-step architecting approach merely illustrates the use of the *Interaction and Interface framework* in practice in order to expose the benefits of this way of reasoning.

4.1 THE 8-STEP ARCHITECTING APPROACH FOR USING THE FRAMEWORK

The following approach is intended to be owned and executed primarily by the system architect. The system architect is not part of any module teams, nor part of any functional teams (i.e. mechanical, electrical engineering), but act as a separate discipline, overlooking the system as it is decomposed and gradually handed over from systems design to module design and embodiment. The primary stakeholders for the system architect are therefore the module owner, the functional leads, and the project owner.

Each layer of decomposition undergoes all 8 steps of the architecting approach, which is aligned with traditional top-down systems thinking. We do however recognize the fact that in reality, products are typically developed in a mix of top-down, bottom-up. Fig. 5 outlines the 8 steps in a graphical way.

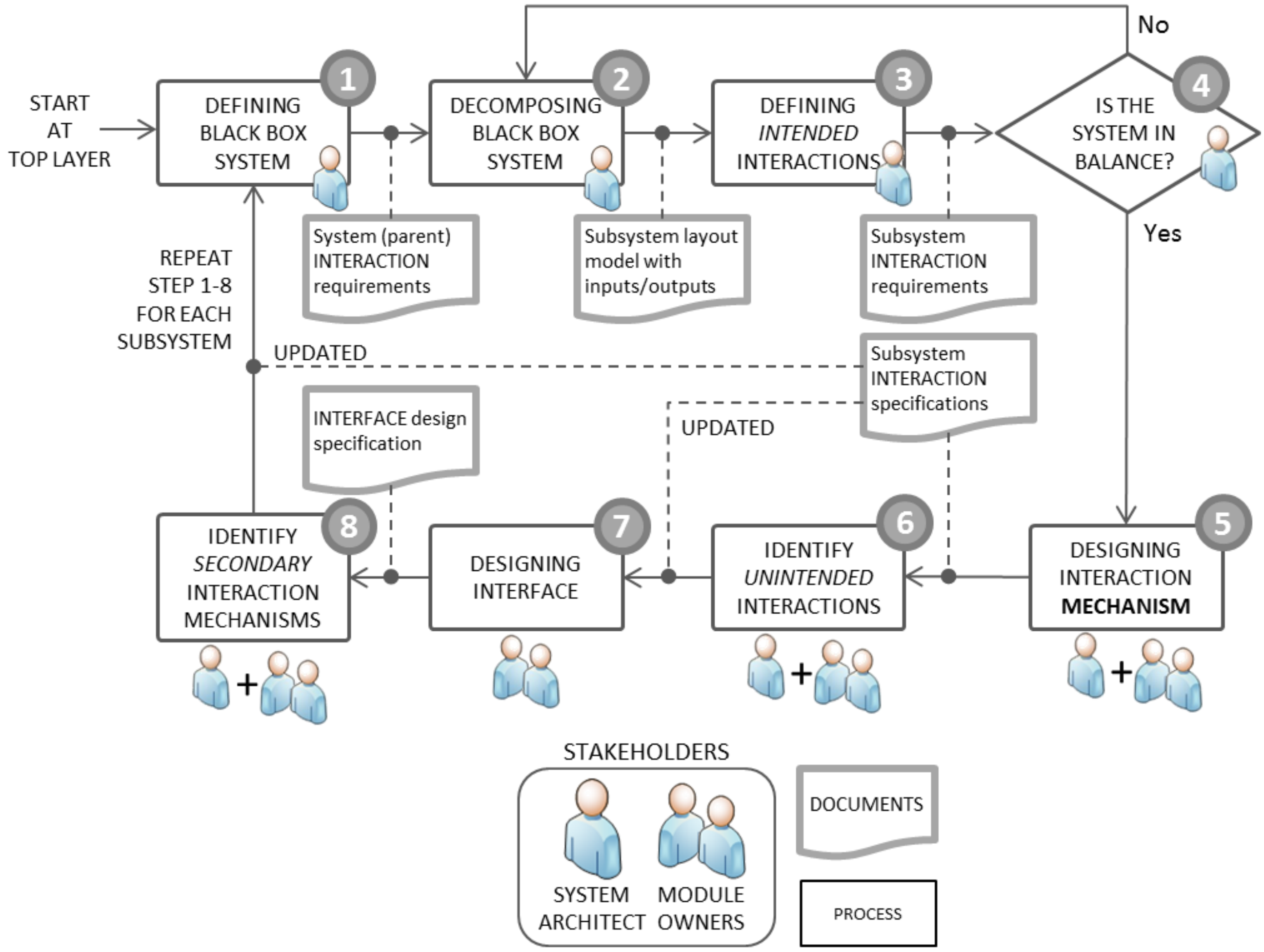


Fig. 5 Illustration of 8-step architecting approach for applying the *Interaction framework*

Step 1 is about defining the system boundary of the Black Box System. Step 2 involves decomposing the system into a sublayer with interacting subsystems. In step 3, the intended INTERACTIONS are defined, followed by a check in step 4 of the consistency between step 2 and 3. Step 5 then proceeds with designing the primary INTERACTION MECHANISMS that facilitate the intended INTERACTIONS. If the INTERACTION MECHANISMS indirectly facilitate unintended INTERACTIONS, these will be captured in step 6. Step 7 then proceed with the design of INTERFACES followed by step 8 where secondary INTERACTION MECHANISMS are identified and documented. Once having completed this approach for one level of decomposition, each subsystem undergoes the same 8-step process until a suitable level has been reached.

The following section will describe step-by-step how the framework is used in principle when synthesizing a product.

4.1.1 Step 1 - Define Black Box System (BBS) and external INTERACTIONS

The approach is initiated by first defining the system boundary, which scopes the extent of the system. As such, we do not know what is "inside" the system, but rather what is "not inside" the system, i.e. external to the system. We may call this a *Black Box System (BBS)*. A part of defining the BBS has to do with identifying the intended INTERACTIONS that needs to be transferred and documenting them in an INTERACTION requirements document.

Consider for example the development of a new *hair drying device*. *Drying hair* is the overall function of the product, which may trigger a wish for *Thermal energy transfer* or *Kinetic energy transfer* as the intended INTERACTIONS. A system architect writes a system level INTERACTION requirement, e.g. "The device shall output a *thermal energy* *transfer rate* in the range of 1200W to 2000W to the user's hair". The same is done for the intended input INTERACTIONS. The *Interaction framework* (Table 2) may be used to look up possible INTERACTIONS.

The output of step 1 is a *system level, INTERACTION requirement document*, which is then input to step 2.

4.1.2 Step 2 – Decompose to next level

Next a hypothesis for a first level decomposition of this functional representation of the system is proposed by first

a) Dividing the parent system into subsystems based on function.
b) Assigning the external intended INTERACTIONS to the subsystems
c) Identifying (not defining) which subsystems that interact, e.g. using N^2-diagrams (Kapurch 2007), DSM (Eppinger and Browning 2012), or Interface diagram (Bruun et al. 2014) as tools to map)

For example the *hair drying device* may be decomposed into three subsystems; heating system, flow system, and power control system. These are all functional elements, which INTERACT to comply with the overall functionality of *hair drying*. The output from step 2 is thus a *system model showing the layout with inputs/outputs between subsystems*.

It should be noted here that the *Interaction framework* does not support the system architect in *how to* decompose the system properly but rather supports the system architect in defining the INTERACTIONS and INTERFACES once the decomposition has been performed.

4.1.3 Step 3 – Define *intended* INTERACTION requirements at a subsystem level

Requirements for the assigned subsystem INTERACTIONS (i.e. mass, charge, momentum and energy) must now be budgeted between the subsystems using the *Interaction framework* (Table 2) as a starting point.

The system architect may for example note the intended/allowed *electric potential energy* from the *power control system* to the *heating system*, whereas the *heating system* transfers a *thermal energy* to the *environment*. The requirements should be defined in ranges and be assigned tolerances to allow for variation and possibly multiple modes of the system (Liu et al. 2015). The output is a *subsystem INTERACTION requirements document*.

4.1.4 Step 4 – Perform 'Sanity' check

Before commencing with designing solutions, a 'sanity' check is needed to ensure that the *subsystem INTERACTION requirements* are reasonable and consistent with the *parent system (parent) INTERACTION requirements*, so that mass or energy does not accumulate/disappear inside the system if it is not intended, e.g. a pump must have the same input and output mass while a battery may periodically store input energy, and thus do not have an output. This check is done by "simulating" the system either by hand or by modeling the system in a suitable IT-software tool. The system must be in equilibrium depending on the operating condition.

4.1.5 Step 5 – Design the INTERACTION MECHANISM

In order to get closer to a design solution the system architect must now define the suitable INTERACTION MECHANISMS for facilitating the INTERACTIONS. This is done by using the *Interaction framework* (Table 2) and identifying the primary INTERACTION MECHANISMS.

In order to characterize the relationship between the INTERACTION MECHANISMS and INTERACTIONS, the system architect may apply the *INTERACTION specification template* (See appendix A, Table 6-7), which support the system architect in identifying the relevant *characteristics of the chosen INTERACTION MECHANISM*.

By means of this template, the system architect can look up the characteristics of the chosen INTERACTION MECHANISM in relation to each facilitated INTERACTION. In the *hair drying device example*, three INTERACTIONS may be considered as significant for the air flow (INTERACTION MECHANISM).

Thermal energy transfer rate = area of the flow X velocity of the flow X volumetric heat capacity X temperature

Mass transfer rate = area of the flow X velocity of the flow X volume mass density

Momentum transfer rate = area of the flow X (velocity of the flow)^2 *X volume mass density*

As can be seen from these three equations, some of the characteristics recur in several equations, i.e. the underlined characteristics. In this way, the trade-offs becomes explicit to the system architect that adjusting the *area of the flow*, not only influence the *mass flow* but also the *momentum* and *thermal energy transfer rate*, and that changing the *velocity of the flow* has an exponential effect on the *momentum transfer rate*.

It is up to the system architect in collaboration with the module owners to perform these trade-off studies by inserting the characteristics of the various types of bulk MATERIAL flows (e.g. gases, liquids, solids) and come up with a working assumption of a configuration of realistic values in the three equations. By additionally analyzing the requirements using *dimensional analysis* the system architect is able to perform a consistency check thus further reducing ambiguity (Mahajan 2014). The output from this process is a *subsystem INTERACTION specification document.*

4.1.6 Step 6 – Identify *unintended* INTERACTIONS

When choosing the desired INTERACTION MECHANISM the system architect must realize that there are some INTERACTIONS that are facilitated which are not intended - we call them *unintended* INTERACTIONS in line with Ulrich and Eppinger (2012). For example an electric current (*Primary*, MICROSCOPIC MATERIAL transfer) may have the purpose of facilitating electric potential energy (intended INTERACTION) however inherently also creates a magnetic force field around the moving charge capable of facilitating *magnetic potential energy* (unintended INTERACTION). This phenomenon is the reason that Electromagnetic Compatibility (EMC) is an important task in multi-disciplinary development. The principle can however also be used productively in for example wireless inductive charging of smartphones, in which case magnetic potential energy transfer is intended.

An unintended INTERACTION may be more or less detrimental to the system and must therefore be captured and documented immediately after step 5 in order to avoid compatibility issues.

Unintended INTERACTIONS can be tricky because they are not necessarily easily detectable through visual inspection (e.g. EMR). Capturing them may therefore rely heavily on the experience of the system architect or designer.

With the *Interaction framework,* unintended INTERACTIONS are captured by reflecting on all possible INTERACTIONS facilitated by a certain INTERACTION MECHANISM and assessing the significance of their impact. As the design of an INTERACTION MECHANISM is matured, one must revisit the *Interaction framework* to see, if the chosen solution principle amplifies the impact of certain unintended INTERACTIONS. In this way, *experience* as an influencing factor may be diminished.

The challenge is to ensure that both the *intended* INTERACTIONS and INTERACTION MECHANISMS comply with their respective requirements while the *unintended* INTERACTIONS and INTERACTION MECHANISMS do not pose a risk to the functionality of the design. If the chosen design of INTERACTION MECHANISM does not allow for a feasible configuration where all INTERACTIONS and INTERACTION MECHANISMS (intended and unintended) comply with their requirements, the system architect must backtrack and change the requirements or decide on another INTERACTION MECHANISM for facilitating the intended INTERACTION. At this stage in the development the cost of rework is however marginal compared to later stages of product development.

Any unintended INTERACTIONS, which is discovered in step 6, needs to be added to the *subsystem INTERACTION requirement document* as well as to the *INTERACTION specification document*. The *updated* subsystem INTERACTION specification document is an output from this process. Including in this document are INTERFACE *conditions* that follow the choice of INTERACTION MECHANISM, which act as design input for the INTERFACE design process.

4.1.7 Step 7 – Design the INTERFACE

The *module or component owners* may now proceed with designing the INTERFACE features, which embodies the INTERFACE. The input to the INTERFACE design process is the INTERACTION specification document including INTERFACE conditions. The INTERFACE design process involves conceptualizing various technical solutions for the physical INTERFACE features, which comply with the design input. In this process various virtues of design are balanced like robustness, producibility, design for (dis)assembly, reliability, user experience, quality-feel etc. Finally a design is chosen, which best balance these virtues.

The output from this process is a *subsystem INTERFACE specification document*.

4.1.8 Step 8 – Identify *secondary* INTERACTION MECHANISMS

In step 8, the module owners must realize that the physical embodiment of the INTERFACE features, may introduce *secondary INTERACTION MECHANISMS*, which simply serves the purpose of supporting the INTERFACE in transmitting the primary INTERACTION MECHANISM.

Thus, designing the embodiment of the *INTERFACE features* differently might trigger different *secondary INTERACTION MECHANISMS*. For example, lifting a container with a crane involves designing an INTERFACE between the crane and the container, which is capable of transmitting the primary FORCE needed to overcome the gravitational ('pull') force from the earth. A hook connected to a chain would have a clearly defined area on each INTERFACE feature where the FORCE is transferred. However, if the INTERFACE design involved two bolted plates with several bolts, the primary FORCE, would result in several secondary FORCES acting on each bolt.

These *secondary INTERACTION MECHANISMS* must therefore be balanced with the *primary INTERACTION MECHANISM* in order to ensure compatibility.

Any secondary INTERACTION MECHANISMS, which are discovered in step 8, needs to be added to the *subsystem INTERACTION requirement document* as well as to the *INTERACTION specification document*.

Each subsystem now has a well-defined system boundary with inputs and outputs and solutions for INTERFACE design. All input and output INTERACTION MECHANISMS (primary and secondary) and INTERACTIONS (intended and unintended) are balanced and collectively comply with the system level (i.e. parent system) INTERACTION requirements document.

The architecting process is continued by decomposing to the next layer and repeating step 1-8 until a satisfactory level of decomposition has been reached.

5 DEVELOPMENT OF SUPPORT TOOL

In order to support step 3 and 5 of the 8-step architecting approach, a simple hands-on tool called the *Interaction Specification Wheel (ISW)* has been developed. The purpose of the tool is to support a system architect in making *complete* and *unambiguous* INTERACTION requirements and specifications – both in terms of classifying any type of INTERACTION in a complex, multi-technological system but also to specify them correctly. In essence, the ISW is a vehicle for the *Interaction framework*, which presents Table 2 and appendix A of the *Interaction framework*, in a single handy format.

5.1 TOOL FOR MAKING IT EASIER TO USE THE FRAMEWORK

The ISW is inspired from traditional calculation wheels typically used in electrical and nuclear engineering. See fig. 6. It has two sides to it; a front side which supports the user in classifying a particular interaction according to an interaction mechanism and back side which supports the user in setting the requirements for the associated INTERACTION and specify the necessary design parameters.

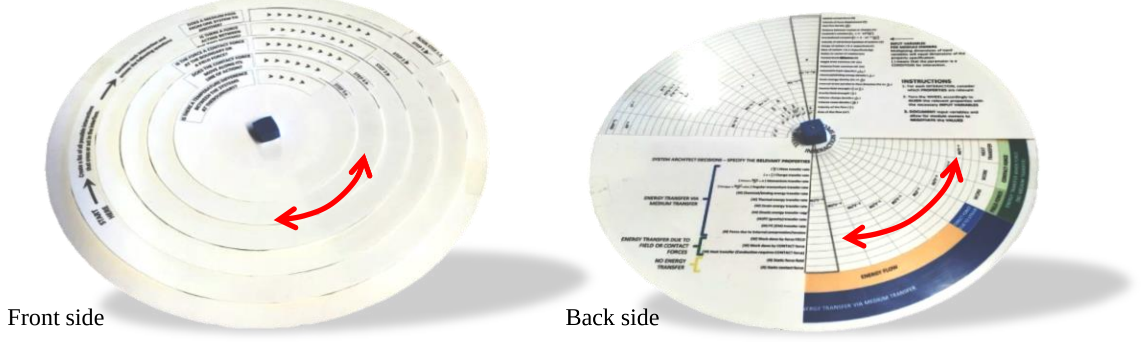


Fig. 6 Prototype of the Interaction Specification Wheel (ISW) having two sides; the front side, which supports the system architect in classifying an INTERACTION and the backside, which supports the system architect in specifying the INTERACTION

Both sides use a principle of rotating discs to guide the system architect to reflect in a certain order. The wheel design was chosen because of its simplicity in representing complex information and its ability to calculate results based on decisions that the system architect makes by turning the discs. Also its handiness in terms of being readily available next to the system architect's keyboard was favorable.

The front side supports the analysis of an existing system by supporting the system architect in classifying a given INTERACTION MECHANISM by asking a series of consecutive questions. Each question can have two outcomes, one of which always leads to the next inner rotating disc, and thereby the next question. The other answers lead to the back side. Answering these questions will ensure that all imaginable INTERACTIONS are classified according to the framework. Over time, once the system architect has gotten used to classifying INTERACTION MECHANISMS using the decision wheel on the front side, they may find themselves only using the back side, which has all of the information necessary to specify INTERACTIONS and INTERACTION MECHANISMS. In that way, the ISW supports the learning curve effect of its users.

The back side contains a complete list of all the INTERACTIONS that are associated with the various INTERACTION MECHANISMS. By turning and aligning the transparent top layer (ruler) with the desired INTERACTION, the wheel tells you which characteristics about the INTERACTION MECHANISM are needed to specify the INTERACTION. By setting requirements for the flow properties, one can compare input and output values to a system, and make sure they are balanced. Given the laws of conservation of momentum and energy, momentum and energy cannot change in a system, unless it is affected by an INTERACTION from outside the system. If the numbers do not add up, it may provoke the system architect to reflect upon losses or accumulations in the system, like heat loss through friction or lack of insulation, chemical energy decay or accumulation etc.

5.2 EVALUATION OF THE INTERACTION FRAMEWORK AND INTERACTION SPECIFICATION WHEEL (ISW)

In order to evaluate the effects of the *Interaction Framework* and *ISW* on practice, five user tests were setup. The test setup was designed to verify whether the *Interaction framework* and ISW live up to the input requirement, see section 1.2.1 Research Method.

The following list summarizes the test setup. For information on reasons behind this particular setup we refer to (Parslov et al. 2016).

- Five individual user tests
- Five test participants (TPs)
    - Area of technical expertise

    - Primary: Mechanical, Electrical,
    - Secondary: Mechatronics, Software, Fluid Mechanics, Thermal, Hydrography, Ultrasound, Systems Engineering, software
  - Years of industrial experience (Listed from TPs 1 through 5)
    - 8, 9, 12, 33, and 40 years of experience.

While a total of five user tests are not enough to conclude with statistical confidence, they may provide an indication of the effects, which can then be further investigated.

5.2.1 Test method

A decomposed *hair dryer* was used as a test case product. See fig. 7.

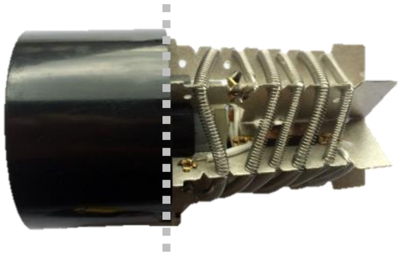


Fig. 7 The blow module (left) and heating module (right) from a hair dryer where used as case product to test the *Interaction framework* and ISW. The grey dotted line represents the interface across which various interactions occur

As part of the test, the TPs were asked different questions to remove sources of error. As such, all TPs *totally agreed* with the statement that *they are familiar with the products mode of action*.

Each TP was individually tested and given three tasks related to a specific INTERFACE in the case product, see fig. 7, grey dotted line:

1. *List all INTERACTIONS that pass or act in that boundary (intended and unintended) given the physical sample in front of you.*
2. *Specify the requirements for two of those INTERACTIONS.*
3. *Specify the related characteristics of the INTERACTION MECHANISMS that the two module owners can negotiate between them.*

These three steps were completed first without prior introduction to the framework and ISW. Secondly they were introduced to the framework and ISW and asked to complete the same tasks. For the sake of this evaluation we are only interested in results of the second and third task. For results of the first task we refer to (Parslov et al. 2016).

After each round of completion the TPs were asked to rate whether they found it easy to identify and specify the INTERACTIONS. After the second round, they were asked whether they thought the framework and ISW *improved* their ability to specify the INTERACTIONS. Although these kinds of qualitative evaluations should be taken with a grain of salt, it was the author's impression that their answers were genuine and honest.

5.2.2 Test results

The tests *indicate* a positive effect of the *Interaction framework* and ISW in terms of creating less ambiguous INTERACTION requirements and specifications. See Table 5. An analysis of the dimensions of the TPs specifications both before and after the introduction to the *Interaction framework* and ISW, shows a significant improvement in terms of creating consistent specifications. Also, 4 out of 5 TPs *agreed* or *totally agreed* that the *Interaction framework* and ISW improved their ability to specify INTERACTIONS.

Due to the limited number of tests performed, we can not conclude with statistical confidence that the framework and the tool will work in general. There is therefore a need for further tests to verify the applicability of the proposed support.

Table 5 Results of five individual tests. Generally the test shows a positive effect in relation to the requirement for the *Interaction framework* and ISW

ID#	Requirement	Results	Conclusion	Comments
1	The **framework and tool** shall lead to a **less ambiguous specification** of the INTERACTIONS across different technical disciplines, than could be achieved without the support.	4/5 of the TPs *agreed* or *totally agreed* that the framework *improved* their ability to specify interactions. 1/5 got more confused. A dimensions analysis also shows that the consistency of specifications improved significantly.	On the basis of these five user tests, and with reservations of a lack of statistical confidence, we can conclude that the Interaction framework and ISW proved *applicable* in terms of significantly increasing the consistency of the TPs INTERACTION specifications as well as *useful*, due to their positive responses.	There is a need for further verification in a real-world, multi-disciplinary project.

It can also be concluded that implementing the framework in practice will require training of users in order to harness the full potential.

6 DISCUSSION OF THE INTERACTION FRAMEWORK IN CONTEXT OF ENGINEERING DESIGN

This discussion revolve around the objective set up for this research as well as some of the issues that are inherently part of product development in any company; is it really feasible to consider all INTERACTIONS and INTERFACES with equal detail and respect? Are there some INTERACTIONS in a system that deserves greater attention? If so, how do you know which ones and how much attention they deserve?

6.1 AMBIGUITY IN RELATION TO SPEAKING THE SAME LANGUAGE

An important objective for this research has been to *reduce ambiguity* during the decomposition of a system. The ambiguity arises when people from different engineering domains communicate with each other using terms like *interaction* and *interface* but without a declaration of how they interpreted the terms. Imprecise communication between co-working engineers may therefore lead to misinterpretations and ultimately rework.

During the evaluation of the framework, the five TPs were asked to speak about what the term *interaction* means to them, as a preparation for the test. As expected, the perceptions across the five TPs reflected a somewhat coherent picture of the *purpose* of an interaction, but a rather inconsistent picture of the *nature* of the interactions. Statements expressed by TPs: *"kind of an interface, where two things meet and interact", "transmission of energy", "Material flow, current, voltage, touch surface", "when two interfacing units has some various levels of properties that needs to match"*.

The introduction of the *Interaction and Interface framework*, allows for a rigorous distinction between an INTERACTION and an INTERFACE. Also, the 8-step architecting approach presents a clear division between the design of INTERACTIONS, the INTERACTION MECHANISMS that facilitate them, and the design of an INTERFACE. With support of the ISW, a common set of equations may be set up that bridge the language barrier between the engineering disciplines.

6.2 COMMON ISSUES IN PRODUCT DEVELOPMENT

We discuss here two common issues in product development; How to assess CRITICALITY (i.e. which interactions should be focused on when cost and time constraints apply) and how to ensure COMPLETENESS (i.e. at what level of detail and concreteness an interaction should be specified).

6.2.1 COMPLETENESS in face of UNCERTAINTY requires ITERATIONS

Assessment of COMPLETENESS of an INTERACTION specification is a measure of how detailed and concrete the description of an INTERACTION is relative to what you know about the system at any given time of the project. Assessment of COMPLETENESS is therefore not considered as an absolute concept but rather as a relative phenomenon that depends on; the level of UNCERTAINTY at any given time in a project as well as the level of *experience* of the assessor (i.e. system architect). These two factors thus collectively become the frame of reference for assessing the COMPLETENESS of an INTERACTION specification.

While the assessment of COMPLETENESS of any INTERACTION specification in any project is relative to UNCERTAINTY and experience, the classification of INTERACTION MECHANISMS as presented in the framework can be considered as COMPLETE in an absolute sense from a physics standpoint. The system architect may therefore rely on the classification as a check list for ensuring that all INTERACTION MECHANISMS and INTERACTIONS have at least been considered, thus enforcing a relatively more COMPLETE INTERACITON specification. Without the framework however, *experience* is the only enabler for assessing the COMPLETENESS of an INTERACTION specification. According to the evaluation presented in Parslov et al. (2016) more experienced engineers capture more interactions first off, than less experienced engineers. However it also showed that the *Interaction framework* seems to minimize the difference of experience thus removing some of the subjectivity and reducing the risk of rework.

Another phenomenon that affects the COMPLETENESS is the fact that you do not understand the significance of a given INTERACTION before understanding the parent level functionality. It is thus the understanding of TOTALITY that guides the decisions about PARTIALITY. Therefore, the INTERACTION specification is only as COMPLETE as the understanding of the parent level permits, e.g. you don't know which INTERACTIONS of the air flow between a blower module and a heater module in a hair dryer are important, if you don't know the higher purpose of the air flow; to whirl up the user's hair and to vaporize the water from the hair, in which case mass flow, momentum flow, kinetic energy and thermal energy transfer rate are important INTERACTIONS – all others can be consciously neglected, i.e. charge transfer rate, strain energy transfer rate, the chemical energy transfer rate etc.

It is because of this phenomenon that the 8-step architecting approach for applying the *Interaction framework*, is a systematic, top-down approach to systems design that allows for ITERATIONS due to the inherent UNCERTAINTY in designing, e.g. as physical solutions are developed, new INTERACTIONS that were not anticipated emerge and are discovered most likely during testing. However with the use of this *Interaction framework* the unintended INTERACTIONS and secondary INTERACTION MECHANISMS are captured early and are added to the INTERACTION model. The tricky thing is that choosing a solution principle at one level of abstraction will most likely lead to added intended and unintended INTERACTIONS at other levels of abstraction. To cope with this requires iterative development and the right organizational setup in terms of ownership and responsibility allocation.

6.2.2 CRITICALITY of INTERACTIONS – where to focus effort

Because of limited resources (e.g. time and money), any company is forced to focus their efforts where there is the highest impact per work effort. The difficult part is to understand *where* that is. In Failure Mode and Effect Analysis (FMEA), R*isk Priority Numbers (RPN)* are used to identify and quantify the risk of potential failure modes. In some sense, the method helps identify the most CRITICAL events to focus on from a bottom up perspective. There are three components in the calculation of RPN values; *severity of a certain event, probability of occurrence and probability of detection.* If we take this definition of RPN and apply it as a definition of CRITICALITY of an INTERACTION MECHANISM, then obviously the *severity* and the *probability of occurrence* are context specific but in terms of *probability of detection* we may imagine some general thoughts about which types of INTERACTION MECHANISMS that are likely to be more critical than others.

MATERIAL transfers and EM MACROSCALE FIELD FORCE EFFECTS are likely to be *more critical* than *EM MICROSCALE FIELD FORCE EFFECTS (physical contact FORCE)* for the following reasons.

As seen in the classification scheme (i.e. Table 2), the number of INTERACTIONS facilitated by MATERIAL transfer is much higher than what is facilitated by a FORCE mechanism. As a consequence, the risk of incompatibility at the INTERFACE might rise because it may be difficult to evaluate which of the INTERACTIONS should be specified and controlled, given that the MATERIAL transfer may affect the receiving system in many different ways.

Also, MATERIAL transfer and MACROSCALE FIELD FORCES are 'global' in nature whereas INTERACTION via a physical contact FORCE is 'local'. INTERACTION MECHANISMS which are not confined by space (i.e. 'global') may affect many more subsystems and are therefore much harder to detect and capture. They may appear as a result of a decision made somewhere completely else in the system.

While INTERACTIONS via FORCE seem less critical than transfers of MATERIAL and MACROSCALE FIELD FORCES, the superimposition principle may easily be forgotten resulting in unintended INTERACTIONS, unanticipated behavior of the system and ultimately rework. Future research might investigate the possibility to apply this notion of CRITICALITY and combine it with methods like Design for Variety by Martin and Ishii (2002) or RPN values from FMEA to create a map of CRITICAL INTERACTION MECHANISMS that deserve detailed attention.

In real-world projects there are all sorts of factors that may be drivers or inhibitors of implementing new ways of working; level of complexity of the product, the size of the project development team, the risk of failing, the risk of delay, cost, micro-political agendas etc. The influence of all these factors must be studied more thoroughly in a real-world case study in order to increase the chances of implementing the *Interaction and Interface framework* in industry. Teaching it at universities may be another, but more long-term way to disseminate it into industry. The powerful thing about the *Interaction and Interface framework* is however, first and foremost that it's a *mindset for reasoning* about *interactions* in a more productive manner. From there on, it may be applied on an individual or organization level and can be rolled out to any level of detail, with more or less rich information.

7 CONCLUSION

This paper builds on a first principle, physics based *Interaction framework* developed to create a common language across any technical discipline that will reduce ambiguity during the architectural decomposition of complex systems. Four contributions are presented in which the two former extends the *Interaction framework* by further qualifying the definition of INTERACTIONS and INTERFACES to the engineering design domain. The two latter contributions are a 8-step architecting approach and a tool called the Interaction Specification Wheel (ISW) to support the application of the *framework* in practice.

The *Interaction and Interface framework* and ISW are evaluated in five individual user tests. The evaluation indicates a positive effect on the test participant's ability to reduce ambiguity in the INTERACTION requirements and specifications. Five user tests does not allow for statistical confidence, however the positive results indicate the need for further research on the implications of the *Interaction and Interface framework* to practice for example in a case study.

Reducing ambiguity in the definition, documentation, and communication of interactions and interfaces in engineering design may significantly improve competitiveness for companies developing complex, multi-technological products, by allowing concurrent engineering, reduced rework and shortened time-to market for new product development.

8 REFERENCES

Albers A, Wintergerst E (2014) The contact and channel approach (C&C2-A): relating a system's physical structure to its functionality. In: An Anthology of Theories and Models of Design. Springer, pp 151–171

Bonnema G, Veenvliet K, Broenink J (2015) Systems Design and Engineering: Facilitating Multidisciplinary Development Projects. CRC Press

Bruun HPL, Mortensen NH, Harlou U (2014) Interface diagram: Design tool for supporting the development of modularity in complex product systems. Concurr Eng 22:62–76. doi: 10.1177/1063293X13516329

Buede DM (2012) The Engineering Design of Systems: Models and Methods, 2nd edn. Wiley

Eppinger S, Browning T (2012) Design structure matrix - methods and applications. MIT Press

Fotso AB, Rettberg A (2012) State of the art for mechatronic design concepts. In: Proceedings of 2012 IEEE/ASME 8th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications. IEEE/ASME, pp 232–240

Grady JO (1994) System integration. CRC press

Haskins C, Forsberg K, Krueger M (2006) Systems Engineering Handbook. INCOSE

Hubka V, Eder WE (1988) Theory of Technical Systems, 2nd edn. Springer-Verlag, New York, NY

Jarratt TAW, Eckert CM, John P (2004) Providing an overview during the design of complex products. 239–258.

Kapurch SJ (2007) NASA Systems Engineering Handbook. NASA Spec Publ 360.

Liu C, Hildre H, Zhang H, Rølvåg T (2015) Conceptual design of multi-modal products. Res Eng Des 26:219–234.

Liu S (2013) Design Controls. In: FDA. http://www.fda.gov/downloads/Drugs/DevelopmentApprovalProcess/SmallBusinessAssistance/UCM377644.pdf. Accessed 6 Feb 2016

Mahajan S (2014) The Art of Insight in Science and Engineering: Mastering Complexity. MIT Press

Martin M V, Ishii K (2002) Design for variety : developing standardized and modularized product platform architectures. Res Eng Des 13:213–235. doi: 10.1007/s00163-002-0020-2

Parslov JF, Gerrard B, Mortensen NH (2016) Understanding Interactions in Complex Multi-Technological Products: A First Principles, Physics-based Theoretical Framework. Res Eng Des 32.

Parslov JF, Mortensen NH (2015) Interface definitions in literature: A reality check. Concurr Eng. doi: 10.1177/1063293X15580136

Ulrich KT, Eppinger SD (2012) Product design and development. McGraw-Hill Higher Education, New York

Wheatcraft LLS (2010) Everything you wanted to know about interfaces, but were afraid to ask. Compliance Automation

Zheng C, Duigou J Le, Bricogne M, Eynard B (2016) Multidisciplinary Interface Model for Design of Mechatronic Systems. Comput Ind 76:24–37.

APPENDIX A

Table 6 INTERACTION specification template – Support for specifying the relevant characteristics of FORCE-based INTERACTION MECHANISMS with respect to INTERACTIONS.

Characteristics of INTERACTION MECHANISM	SI Unit	Dimensions	*Thermal energy transfer rate*	*Kinetic energy transfer rate*	*Strain energy transfer rate (work)*	*Potential gravitational energy transfer rate*	*Potential electric energy transfer rate*	*Potential magnetic energy transfer rate*
INTERACTIONS facilitated by FORCE — SI Unit			J/s	J/s	J/s	J/s	J/s	J/s
INTERACTIONS facilitated by FORCE — Dimensions			ML^2T^{-3}	ML^2T^{-3}	ML^2T^{-3}	ML^2T^{-3}	ML^2T^{-3}	ML^2T^{-3}
Area of the flow	m^2	L^2	✓	✓				
Mass of interacting systems respectively	kg	M_1M_2				✓		
Charge of interacting systems respectively	C	Q_1Q_2					✓	
Velocity of attraction/repulsion of systems if work	$\frac{m}{s}$	LT^{-1}				✓	✓	
Gravitational constant $\left(G = 6 \cdot 10^{-11}\frac{Nm^2}{kg^2}\right)$	$\frac{Nm^2}{kg^2}$	$L^3M^{-1}T^{-2}$				✓		
Coulomb's constant $\left(k_e \approx 9 \cdot 10^{9}\frac{Nm^2}{C^2}\right)$	$\frac{Nm^2}{C^2}$	$ML^3Q^{-2}T^{-2}$					✓	
Distance between masses or charges	m	L				$(✓)^2$	✓	
Heat flux density	$\frac{W}{m^2}$	MT^{-3}	✓					
Wave energy flux density (= Intensity of radiation)	$\frac{W}{m^2}$	MT^{-3}		✓				
Velocity of force displacement (if work is done)	$\frac{m}{s}$	LT^{-1}			✓			
Applied contact force (* = force is conditional)	N	MLT^{-2}	✓*		✓			
Magnetic moment	Am^2	IL^2						✓
Magnetic flux density transfer rate (= Field strength)	Tesla / t	$MT^{-3}I^{-1}$						✓

APPENDIX A - continued

Table 7 INTERACTION specification template – Support for specifying the relevant characteristics of MATERIAL transfer based INTERACTION MECHANISMS with respect to INTERACTIONS.

Characteristics of INTERACTION MECHANISM	SI Unit	Dimensions	Mass transfer rate	Charge transfer rate	Translational momentum transfer rate (*Force*)	Angular momentum transfer rate (*Torque*): *Translational angular momentum (orbit)*	Angular momentum transfer rate (*Torque*): *Rotational angular momentum (spin)*	Energy – *Intrinsic energy*: *Chemical energy transfer rate*	Energy – *Intrinsic energy*: *Thermal energy transfer rate*	Energy – *Intrinsic energy*: *Strain energy transfer rate*	Energy – *Extrinsic energy*: *Kinetic energy transfer rate*	Energy – *Extrinsic energy*: *Potential gravitational energy transfer rate*	Energy – *Extrinsic energy*: *Potential electric energy transfer rate*	Energy – *Extrinsic energy*: *Potential magnetic energy transfer rate*	Force due to internal strain
SI Unit			kg/s	C/s	N	Nm	Nm	J/s	J/s	J/s	J/s	J/s	J/s	J/s	N
Dimensions			MT^{-1}	I	MLT^{-2}	ML^2T^{-2}	ML^2T^{-2}	ML^2T^{-3}	ML^2T^{-3}	ML^2T^{-3}	ML^2T^{-3}	ML^2T^{-3}	ML^2T^{-3}	ML^2T^{-3}	MLT^{-2}
Area of the flow	m^2	L^2	✓	✓	✓	✓		✓	✓	✓	$\frac{1}{2}$(✓)	✓	✓	✓	✓
Velocity of the flow	$\frac{m}{s}$	LT^{-1}	✓	✓	(✓)^2	(✓)^2		✓	✓	✓	(✓)^3	✓	✓	✓	
Volume mass density	$\frac{kg}{m^3}$	ML^{-3}	✓		✓	✓					✓	✓			
Volume charge density	$\frac{C}{m^3}$	ITL^{-3}		✓									✓		
Gravity field strength	$\frac{m}{s^2}$	LT^{-2}										✓			
Electric field strength	$\frac{N}{C}$ or $\frac{V}{m}$	$MLI^{-1}T^{-3}$											✓		
Internal stress parallel to flow direction (*Pa*)	$\frac{N}{m^2}$	$ML^{-1}T^{-2}$													✓
Strain energy density (*Pa*)	$\frac{N}{m^2}$	$ML^{-1}T^{-2}$								✓					
Chemical (binding) energy density	$\frac{J}{m^3}$	$ML^{-1}T^{-2}$						✓							
Volumetric heat capacity	$\frac{J}{m^3 \cdot K}$	$ML^{-1}T^{-2}\theta^{-1}$							✓						
Distance from common reference point	m	L											✓		
Height from common reference point	m	L										✓			
Temperature	(K)	θ							✓						
Radius to center of rotation	m	L				✓									
Angular velocity	$rad \cdot T^{-1}$	T^{-1}					(✓)^2								
Moment of inertia	$kg \cdot m^2$	ML^2					✓								
Magnetization	$\frac{A}{m}$	IL^{-1}												✓	
Magnetic flux density	$\frac{kg}{A \cdot s^2}$	$MI^{-1}T^{-2}$												✓	

DTU Mechanical Engineering
Section of Engineering Design and Product Development
Technical University of Denmark

Produktionstorvet, Bld. 426
DK-2800 Kgs. Lyngby
Denmark
Phone (+45) 4525 6263
Fax (+45) 4593 1577
www.mek.dtu.dk
ISBN: 978-87-7475-447-3

DCAMM
Danish Center for Applied Mathematics and Mechanics

Nils Koppels Allé, Bld. 404
DK-2800 Kgs. Lyngby
Denmark
Phone (+45) 4525 4250
Fax (+45) 4593 1475
www.dcamm.dk
ISSN: 0903-1685